

**Universitat de Lleida**

**Escola Politècnica Superior**

**Grau en Enginyeria Electrònica Industrial i  
Automàtica**

**Treball Final de Grau**

**Procediment d'equilibrat  
de ventiladors de sostre per  
implementar en microcontroladors**

Autor: Manel Vidal Antolín

Director: Francisco Clariá

Juliol 2015



“Everything is determined,  
the beginning as well as the end,  
by forces over which we have no control.  
It is determined for the insect, as well as for the star.  
Human beings, vegetables, or cosmic dust,  
we all dance to a mysterious tune,  
intoned in the distance by an invisible piper.”

“You never fail until you stop trying.”

Albert Einstein

## Agraïments

Al meu tutor, Francesc Clariá, per la seva dedicació tots aquests anys, per la seva empremta en la titulació i especialment en la primera promoció de graduats en enginyeria electrònica industrial i automàtica de la UDL. A més dels continguts acadèmics, ens has ensenyat a posar passió en allò que fem.

A Rafael Jordà, responsable dels cursos de formació que imparteix una empresa del sector de l'equilibrat. A partir d'ara compartirem el fet d'haver dedicat el treball de final de carrera al món de les vibracions. No tinc el gust de conèixer-lo personalment, i ja m'ha ajudat més del que li he demanat. En el camí d'obtenir la titulació, d'ell he après la germanor de la professió.



## RESUM

El projecte s'ha centrat en el desenvolupament d'un sistema d'equilibrat de ventiladors de sostre, que posteriorment pugui ser integrat en un microcontrolador.

Per a aquest propòsit, es capturen les dades de la cinemàtica del ventilador, analitzant la seva dinàmica, i es calculen la massa i la posició dels contrapesos necessaris per a corregir els desequilibris del rotor i les aspes.

Per al desenvolupament del sistema d'equilibrat s'utilitza Matlab. Mitjançant l'anàlisi gràfic de les acceleracions del ventilador es desenvolupen els algorismes matemàtics per seleccionar les dades òptimes que permetran calcular la massa i la posició dels contrapesos amb major precisió.

## Paraules clau:

Equilibrat, balanceig, vibracions, ventiladors, anàlisi, filtrat i tractament de senyal, microcontrolador.

## ABSTRACT

The project has focused on developing a balanced system of ceiling fans, which can afterward be integrated into a microcontroller.

For this purpose, it is captured kinematic quantities of a fan, analysing its dynamic, and calculated the mass and position of counterweights required to correct the imbalances of the rotor blades.

To develop the balanced system, it is used the platform Matlab. By chart analysis of the fan accelerations, will develop mathematical algorithms that will allow to select optimal data to calculate the mass and position of counterweights with greater precision.

## Keywords:

Balanced, balancing, vibration, fans, analysis, filtering and signal processing, microcontroller.

## ÍNDIX GENERAL

I.	INTRODUCCIÓ .....	7
A.	<i>Antecedents i Motivació .....</i>	<i>7</i>
B.	<i>Objectiu .....</i>	<i>8</i>
C.	<i>Abast .....</i>	<i>8</i>
II.	Estat de l'art .....	9
A.	<i>Equilibrats industrials .....</i>	<i>9</i>
1.	Mètodes d'equilibrat .....	11
2.	Maquinària .....	12
3.	Metodologia d'equilibrat .....	13
4.	Procediment d'equilibrat .....	14
B.	<i>Equilibrat de ventiladors de sostre .....</i>	<i>15</i>
III.	EQUILIBRAT DE VENTILADORS DE SOSTRE .....	16
A.	<i>Fonament teòric .....</i>	<i>16</i>
1.	Suposicions per simplificar el model teòric .....	17
2.	Determinació de la "massa equivalent" del conjunt .....	17
3.	Correcció del desequilibri amb contrapesos .....	18
B.	<i>Materials .....</i>	<i>21</i>
1.	Sensor de rotació .....	21
2.	Sensor acceleròmetre .....	22
3.	Microcontrolador .....	23
4.	Connexions del conjunt .....	24
C.	<i>Desenvolupament del procediment d'equilibrat .....</i>	<i>25</i>
1.	Elaboració del software del microcontrolador i de Matlab. ....	25
2.	Captura i emmagatzematge de les dades .....	27
3.	Anàlisi previ per al tractament de la senyal .....	28
4.	Elecció de la freqüència de mostreig .....	33
5.	Filtrat .....	34
6.	Monitorització i anàlisi de les dades de les vibracions .....	37
7.	Anàlisi de la senyal (elecció de valors òptims per al càlcul) .....	47
8.	Càlcul del període, l'amplitud i la fase de l'acceleració .....	53
9.	Càlcul dels contrapesos per a l'equilibrat (massa i posició) .....	56
10.	Millora de la precisió i vibracions residuals .....	58
IV.	CONCLUSIONS .....	60
V.	CONTINUÏTAT DEL TREBALL .....	61
VI.	RELACIÓ DE FONTS I BIBLIOGRAFIA CONSULTADA .....	62
VII.	ÍNDIX DE FIGURES .....	64
VIII.	ÍNDIX DE GRÀFIQUES .....	65
IX.	ÍNDIX DE TAULES .....	65

<b>X. ANNEXES .....</b>	<b>66</b>
<b>I. Programari microcontrolador.....</b>	<b>66</b>
<b>II. Funcions creades amb Matlab. ....</b>	<b>83</b>
1. Read_Serial.m .....	83
2. Save_File.m .....	84
3. analisi_senyal.m .....	84
4. fir_2_3_fs110_o197.m .....	90
5. fir_3_4_fs110_o197.m .....	90
6. fir_4_5_fs110_o197.m .....	91
7. phase.m.....	92
8. evolucio_temps.m .....	92
9. calcul_contrapesos.m .....	95
<b>III. Coeficients dels filtres FIR.....</b>	<b>98</b>
<b>IV. Annex IV.....</b>	<b>102</b>
1. Tasques a realitzar.....	102
2. Pressupost .....	103
3. Taules de toleràncies a vibracions residuals .....	104
<b>V. Indicacions per a l'operari .....</b>	<b>107</b>
1. Consideracions i comprovacions prèvies a l'equilibrat .....	107
2. Procediment d'equilibrat.....	107
3. Erros freqüents .....	108

## I. INTRODUCCIÓ

### A. Antecedents i Motivació

La globalització de l'economia i en concret els baixos costos de producció que s'aconsegueixen al que avui dia s'anomena "la fàbrica del món", és a dir, a la Xina, han introduït al mercat una gamma de productes de molt baix cost, però alhora amb uns estàndards de qualitat molt ajustats. Així doncs podem trobar ventiladors domèstics de sostre a partir de 30€. Per reduir costos de muntatge, embalatge, magatzem i transport, el muntatge final se li deixa al usuari.

Com tota màquina rotativa, tindrà vibracions si el centre de masses dels elements en rotació no queda perfectament centrat sobre l'eix de gir. Si es vol corregir, cal posar contrapesos un cop el ventilador està muntant. El problema rau en calcular la massa i la posició dels contrapesos que equilibrin l'aparell. Per fer-ho, serà necessari adquirir les dades de les vibracions i aplicar la tècnica d'equilibrat.

Per trobar la solució òptima caldrà aplicar de forma transversal les diferents matèries del grau en enginyeria electrònica industrial i automàtica. Veurem que una aplicació aparentment senzilla, requereix un tractament multidisciplinari, i alhora, "*l'enginy*" per trobar o crear les eines idònies per triar les millors opcions i anar superant les dificultats molts cops imprevistes.

En aquest TFG el problema a resoldre és trobar un sistema econòmicament viable per equilibrar de forma eficaç i eficient els ventiladors de sostre domèstics. Es pretén evitar les molèsties i els riscos associats a les vibracions que aquests ventiladors solen tenir. Vol ser un projecte pràctic, doncs vol donar solució a un problema real, i alhora, vol ser un projecte acadèmic, doncs requerirà recerca i anàlisis.

Si bé els ventiladors que volem equilibrar són d'ús domèstic, per trobar les millors solucions haurem d'aprofundir en un camp força interessant a la indústria, com és el de les vibracions i el manteniment preventiu de màquines. Podem dir que disposem d'un estat de l'art avançat en l'àmbit industrial, però a uns costos que en fan inviable la seva aplicació domèstica. És per això que actualment, aquests ventiladors, o no s'equilibren, o es fa pel mètode d'assaig error. Un mètode força laboriós i poc precís que no acaba d'eliminar suficientment les vibracions.

L'avanç tecnològic dels darrers anys en el nivell d'integració dels sistemes electrònics amb microprocessadors i la disponibilitat de sensors de precisió i pantalles de visualització a baix cost, ens permet plantejar en aquest treball el repte de portar al camp domèstic una tecnologia fins ara només a l'abast del món industrial.

## B. Objectiu

El projecte es centrarà en desenvolupar un sistema d'equilibrat que pugui ser integrat en un microcontrolador. Un cop dissenyat i provat amb l'ajut d'eines de càlcul i anàlisi de senyal, el següent pas seria aplicar-lo per construir un petit i barat aparell electrònic capaç de mesurar les vibracions d'un ventilador i, amb una petita interfase gràfica, indicar-nos com corregir-les. En el desenvolupament del mètode es tindrà en compte que el futur aparell d'equilibrat pugui ser:

- **Senzill**, que el pugui utilitzar un operari sense necessitat de formació específica. Només guiant-se per un manual o per indicacions d'una interfície com pot ser una petita pantalla.
- **Autònom**, que es pugui alimentar amb bateria i no precisi de connexió a un ordinador. Que per si sol pugui realitzar la captura i emmagatzematge de dades, el tractament de la senyal, el filtrat, els càlculs i tenir una interfície per comunicar-se amb l'usuari per indicar els passos del procés i mostrar per pantalla els resultats.
- **Econòmic**. En aquest sentit intentarem minimitzar el nombre d'elements de hardware i triar aquells que complint les exigències, siguin més assequibles.

## C. Abast

En una primera fase ens servirem d'un microcontrolador i sensors per poder capturar les dades d'un ventilador de sostre i enviar-les a un ordinador. Caldrà fer el muntatge, la programació del microcontrolador i les llibreries del sensor acceleròmetre, les connexions del microcontrolador amb els sensors i amb l'ordinador i també el condicionament de senyal d'un transductor òptic.

La segona fase, més complexa i nucli del treball, és tota la fase d'estudi, proves, detecció de problemes i cerca de solucions òptimes fins a determinar un sistema d'equilibrat de ventiladors de sostre que respongui als objectius plantejats.

En aquest projecte, la funció del microcontrolador és limita a capturar les dades d'acceleracions del ventilador i traspasar-les a l'ordinador per poder-ne fer el corresponent anàlisi. Tot i això, un cop dissenyats els filtres de senyal idonis per al nostre objectiu, s'implementaran al microcontrolador per provar-ne l'aplicabilitat.

Per a fer l'anàlisi i tractament de la senyal ens servirem del software Matlab, el qual ens permet determinar els paràmetres de la senyal, elegir una freqüència de mostreig òptima, experimentar amb diferents tipus de filtrat i convertir una cadena de dades numèriques intel·ligibles en informació gràfica que ens donarà una visió clara del que està passant al ventilador.

En l'apartat 1 de l'annex IV es detallen les tasques que haurem de realitzar. En l'apartat 2 s'inclou un pressupost dels materials utilitzats i la càrrega de treball orientativa.

## II. Estat de l'art

En l'àmbit industrial el problema de les vibracions en màquines rotatives és evident. Una màquina equilibrada pateix menors friccions i fatigues. El consum d'energia serà menor, mentre que la vida útil de les peces, la fiabilitat i la precisió de la màquina serà majors. les vibracions poden ser perilloses per a les màquines o les persones. A determinades freqüències i amplituds es poden transmeten als edificis amb les conseqüents molèsties i risc de fatiga d'algunes estructures.

Si tenim en compte els costos que pot comportar una avaria que aturi una cadena de producció, entendrem que en matèria de "manteniment preventiu" es dediquin esforços i recursos a reduir les vibracions. L'anàlisi dinàmic de màquines rotatives per vibracions ens permet, en funció de la freqüència de vibració, determinar quines peces deteriorades acabarien provocant una avaria. A més, el fet d'analitzar i comparar l'evolució de les vibracions en una màquina al llarg del temps en aportarà una informació molt valuosa sobre el seu estat d'ús i manteniment.

És per això que en aquest àmbit trobem gran varietat de bibliografia, instruments i serveis d'empreses destinades a l'equilibrat de màquines industrials i rotors. També existeixen recomanacions i normes (ANSI, ISO, AFNOR, API, RATHBONE, etc) que determinen les toleràncies a les vibracions en funció de les característiques de la màquina. A l'annex 4, apartat 3, s'adjunten les taules 4, 5 i 6 que indiquen les toleràncies admissibles en funció del tipus de màquina o la freqüència de rotació.

En canvi, per a l'equilibrat de ventiladors de sostre domèstics, en els quals les vibracions no representen un problema greu, veurem que en el millor dels casos s'usa la tècnica de l'assaig error amb contrapesos fins reduir parcialment les vibracions.

### A. Equilibrats industrials

Dins l'àmplia bibliografia sobre fonaments teòrics de l'anàlisi de vibracions en màquines rotatives i de la metodologia d'equilibrat, seran important referència en aquest treball el llibre "*Análisis dinámico de máquinas rotativas por vibraciones*" de Pedro Fraga López (Universidade da Coruña. Servicio de Publicacións, 1998, ISBN 8489694567) i el projecte de final de Carrera "*Estudio de una máquina equilibradora de ejes HOFMANN modelo HL 14-1*" del Departamento de Ingeniería Mecánica, Energética y de Materiales de la Universidad pública de Navarra. Del senyor Juan Martín Santesteban Cizur, en el qual es desenvolupa el procés d'equilibrat de discs i eixos.

Així mateix, pel que fa al procediment d'equilibrat ens basarem en els criteris i recomanacions per a la formació dels seus operaris que ens ha facilitat l'empresa Hofmann-Global, amb seu a Pfungstadt (Alemanya) i amb àmplia experiència en el sector.

*“Les causes de les vibracions provenen dels canvis constants d’una força que modifica contínuament la seva direcció o la seva intensitat. Les fluctuacions d’aquesta força són les que originen les vibracions i les característiques resultants es determinen pel reconeixement de la forma com és generada la força”* (Den Hartog, J.P. “Mechanical Vibrations” Mc-Graw-Hill Book Company, New York)

En una màquina rotativa, les vibracions canviaran d’intensitat i direcció de la mateixa manera amb què les forces que les provoquen modifiquen la seva posició respecte a la màquina, per tant, les vibracions resultants tindran una freqüència que serà funció directa de la velocitat de rotació del component defectuós. Les vibracions produeixen fenòmens dinàmics (amb tendències temporals –interessants d’analitzar des del punt de vista de la prevenció- per la petjada que van deixant en les màquines)

Com a causes principals de les vibracions trobem el desequilibri de masses de les parts giratòries, la desalineació, la ressonància del sistema, les friccions internes, els coixinets defectuosos o les deformacions de l’eix. Les vibracions poden tenir sentit radial i axial. Es podrien fabricar peces amb molt baixa tolerància per evitar-les, però de vegades resulta més econòmic fer peces menys precises i després equilibrar-les.

Les vibracions generen moviments harmònics que habitualment també són periòdics. Per tant, quan analitzem la vibració en el temps, el primer que haurem de fer és passar al domini de la freqüència per determinar l’espectre de la vibració.

Respecte a la senyal relacionada amb les vibracions, ja sigui posició, velocitat, acceleració o forces, caldrà determinar:

- L’amplitud (“magnitud del problema”)
- La freqüència (“origen de la vibració”)
- La fase o angle que pren respecte una referència (“situació del desequilibri”)
- L’amplitud i fase de la resposta obtinguda a les freqüències múltiples ( $n \cdot \omega$ ) de la velocitat angular nominal.
- La forma d’ona

La mesura de les vibracions s’ha de fer pels graus de llibertat del sistema. Així trobem diferents estructures mecàniques i uns models predefinits a l’hora d’analitzar cada cas. Per exemple, una estructura mecànica amb un grau de llibertat pot respondre als models: lliure sense amortiment, lliure amb amortiment o forçat amb amortiment.

Per simplificar el model d’anàlisi de vibracions es sol anar a un disc (que es pot equilibrar treballant sobre un pla) o a un cilindre (on caldrà actuar sobre dos plans) subjectat per dos coixinets. Per analitzar les vibracions caldrà conèixer també les rigideses dels coixinets i la rigidesa de l’eix.

## 1. Mètodes d'equilibrat

Per il·lustrar els conceptes d'aquest apartat reproduïrem algunes figures del treball "Estudi d'una màquina equilibradora d'eixos", de Juan Martín Santesteban Cizur

### a) Equilibrat estàtic

Permet corregir desequilibris en discos afegint i traient material fins a aconseguir situar el centre de masses exactament sobre el centre de rotació. A la Figura 1 observem el model físic i una aplicació pràctica per equilibrar un disc buscant el desequilibri per gravetat.

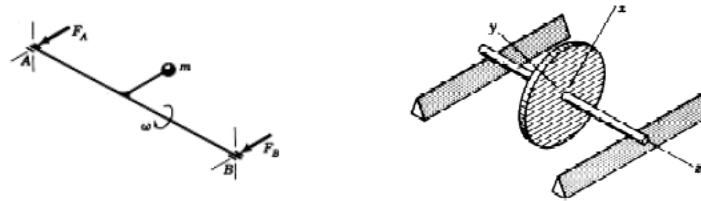


Figura 1: Esquema físic i aplicació d'equilibrat estàtic d'un disc

### b) Equilibrat dinàmic

Quan tenim una distribució de masses sobre un eix. És a dir, ja no parlem d'un disc, sinó d'un cilindre. Podem tenir el sistema estàticament equilibrat, amb el centre de masses sobre l'eix de rotació, però amb una distribució de masses que al girar ens produirà moments d'inèrcia i forces asimètriques sobre els coixinets que produiran vibracions. A la Figura 2 es descriu el model físic d'un cilindre que, estant equilibrat estàticament, té desequilibri dinàmic. A la figura 3 s'observen els efectes del desequilibri estàtic i dinàmic en un cilindre.

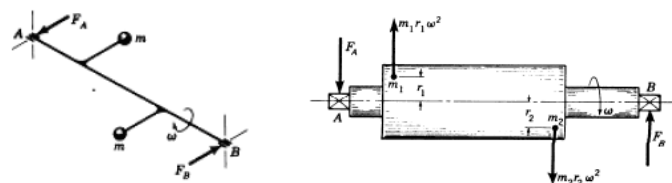


Figura 2: Esquema físic i representació del desequilibri dinàmic en un cilindre

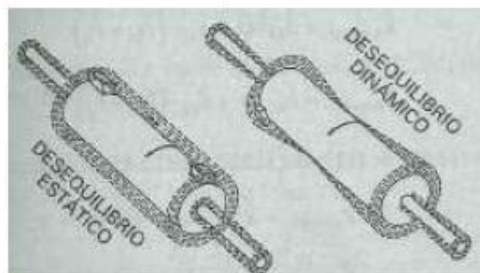


Figura 3: Efectes del desequilibri estàtic i dinàmic en un cilindre



L'equilibrat estàtic (d'un disc) es pot fer per gravetat o inercialment. En el segon cas es fa rodar el disc per mesurar l'amplitud i la fase de les vibracions.

En l'equilibrat dinàmic sempre necessitem el moviment de rotació. L'equilibrat es farà sobre dos plans paral·lels entre ells i perpendiculars a l'eix de rotació. La intersecció dels plans sobre l'eix ens dóna 2 discos sobre els quals s'afegirà o traurà massa per compensar els moments d'inèrcia que genera la distribució irregular de masses en l'eix a equilibrar.

## 2. Maquinària

D'una banda podem trobar màquines portàtils per equilibrat "*in situ*" sense haver de desmuntar el rotor (figura 4) El seu cost pot oscil·lar entre els 1200 i els 8500€ segons les prestacions. D'altra banda també hi ha màquines per equilibrat on cal situar el rotor en un banc de proves (figura 5) En aquest cas els preus poden ser molt variables. Algunes d'aquestes màquines poden fer forats o fressar els elements a equilibrar per distribuir la massa correctament.



Figura 4: Màquines portàtils d'equilibrat in situ



Figura 5: Màquina d'equilibrat en bancada

### 3. Metodologia d'equilibrat

En primer lloc necessitem conèixer les equacions que regiran el moviment. Aquestes dependran dels graus de llibertat, de l'amortiment de la vibració, etc. Després haurem de triar el hardware que ens permeti captar les variables de les equacions. També haurem de tractar la senyal, doncs contindrà sorolls o harmònics que ens emmascararan la senyal que volem. A la figura 6 es mostra la senyal captada per un sensor i la mateixa senyal després del procés de filtrat.

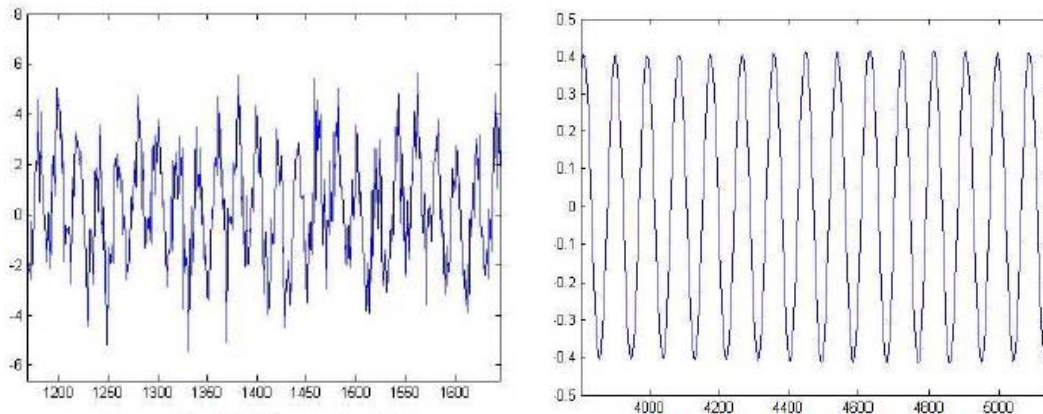


Figura 6: Esquerra: Senyal captada pel sensor. Dreta: Senyal filtrada

Un cop haguem extret de la senyal les dades que necessitem per als càlculs, aplicarem les equacions i haurem de presentar els resultats a l'usuari. A la figura 7 es mostra la interfase d'usuari creada en el projecte del Sr. Juan Martín Santesteban.

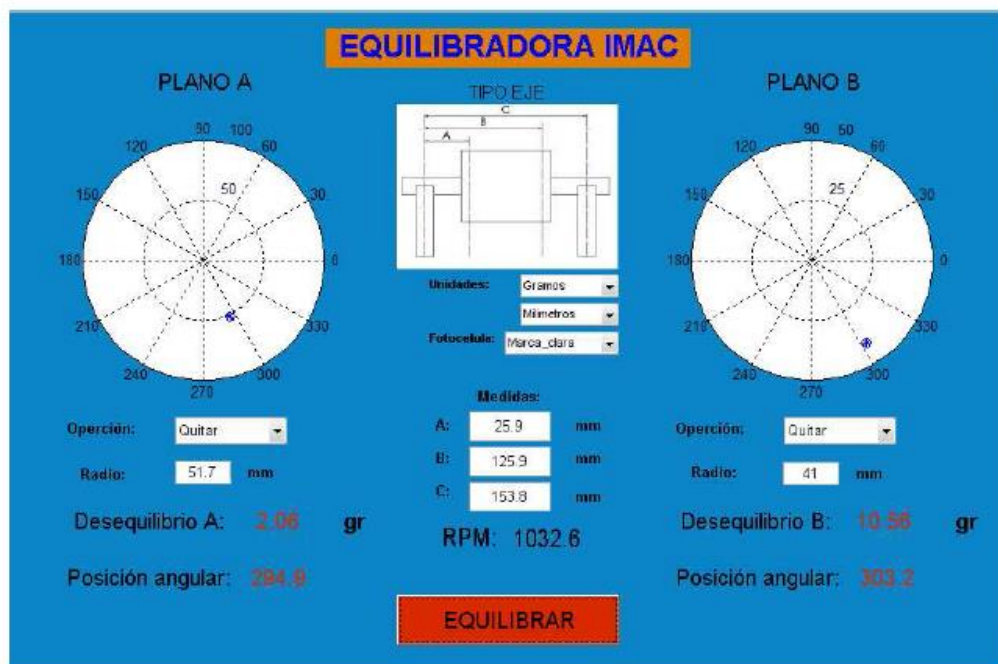


Figura 7: Interfase d'usuari per a equilibradora d'eixos en dos plans

#### 4. Procediment d'equilibrat

L'equilibrat consisteix en controlar o millorar, en cas que sigui necessari, la distribució de masses d'un cos giratori. Per fer-ho podem afegir, treure o desplaçar masses. Aplicarem una correcció "polar" si disposem de tota una circumferència per modificar la massa. Quan només puguem aplicar la correcció en posicions angulars predefinides, parlarem de "correcció en components".

A l'annex V es faciliten indicacions d'operari sobre el procediment d'equilibrat. A l'apartat 1 les comprovacions i consideracions prèvies. A l'apartat 2, els passos a seguir durant el procés d'equilibrat. A l'apartat 3 els errors habituals en el procés.

A la figura 8 trobem l'esquema d'ubicació dels sensors d'una màquina d'equilibrat portàtil.

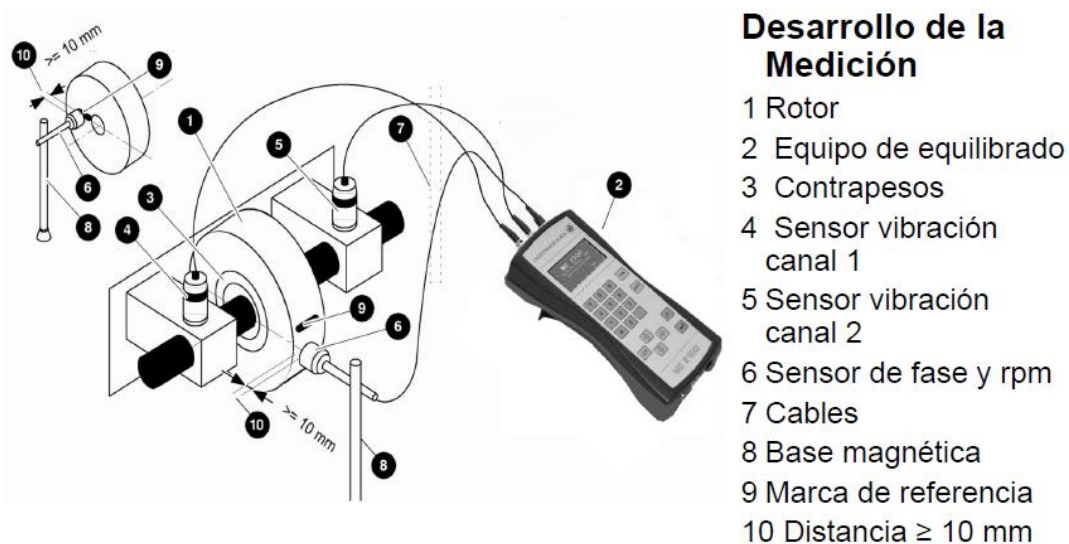


Figura 8: Equilibrat d'un disc amb una màquina portàtil

En el nostre cas, per equilibrar un ventilador de sostre, considerarem un sistema amb 2 graus de llibertat (desplaçament en el pla horitzontal) i aplicarem un equilibrat estàtic inercial, realitzat *in situ*, amb una correcció en components (sobre les aspes).

El model no correspon a un disc que gira subjectat per dos coixinets, dons només està subjectat pel sostre. De fet podem dir que estem intentant equilibrar un pèndul. El que manté en una posició relativament fixa és la gravetat. Serà aquesta, més les friccions de l'ancoratge, les que generant les forces de reacció a les vibracions. Sabem que com menys es desplaça de la vertical, -com menys vibri, menors seran aquestes forces. Per tant, encara que reduïm molt els desequilibris, com també es redueixen les forces de reacció, sempre ens quedarà una vibració residual.

## B. Equilibrat de ventiladors de sostre

Pel que fa a ventiladors de sostre, només hem trobat alguns tutorials a internet basats en la tècnica de l'assaig-error. Algunes empreses comercialitzen "kits" d'equilibrat amb una pinça per fer proves i uns contrapesos per enganxar a les aspes.

La tècnica consisteix en anar col·locant la pinça en diferents aspes fins a determinar, per observació directa, l'aspa on aquesta pinça redueix més la vibració. Posteriorment, com es s'indica a la figura 9, s'anirà desplaçant a diferents distàncies buscant un millor equilibrat. Finalment es substituirà la pinça per un contrapès que s'adherirà a la part superior de l'aspa.

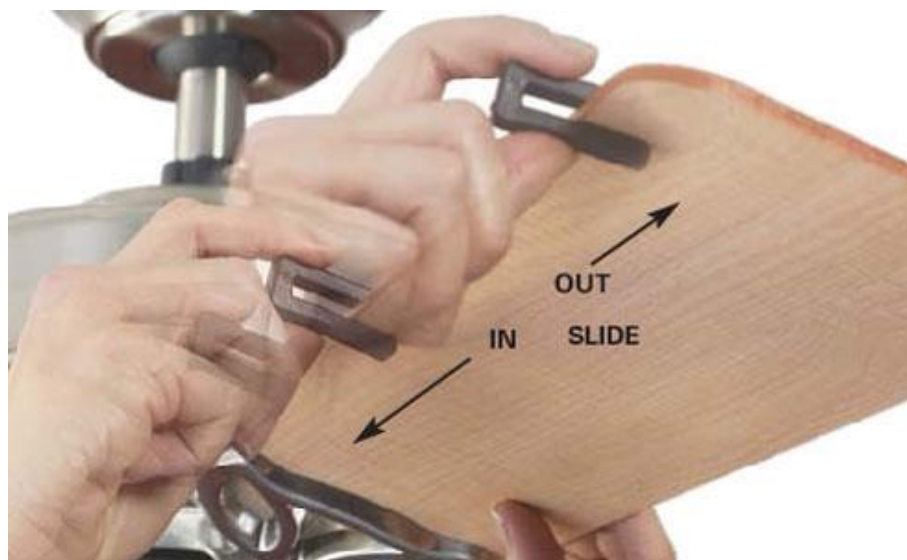


Figura 9: Equilibrat per assaig-error d'un ventilador de sostre

És evident que aquest sistema, a part de ser laboriós, és poc precís. Però quan parlem de ventiladors de sostre que es poden trobar al mercat a partir dels 30€, entenem que no es dediquin els mateixos recursos que a l'equilibrat industrial.

### III. EQUILIBRAT DE VENTILADORS DE SOSTRE

#### A. Fonament teòric

Les vibracions que podem observar en les màquines amb peces giratòries, com és el cas dels ventiladors, són pel fet que tenen desplaçat el centre de masses respecte al centre de gir.

En la figura 10 es descriu un model esquemàtic d'un disc que gira amb el centre de masses desplaçat respecte a l'eix de gir. Considerarem el desequilibri de masses del conjunt giratori com una massa puntual **M**, la qual gira respecte al centre de gir **O**, descrivint una trajectòria circular a una velocitat angular  $\omega$ .

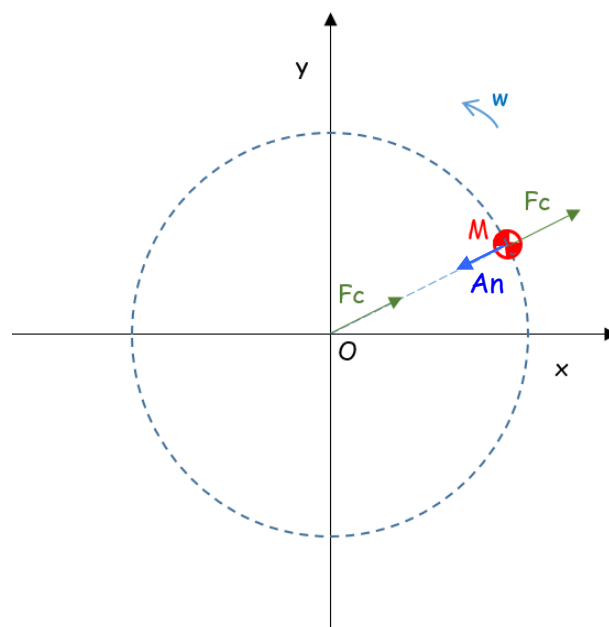


Figura 10: Representació esquemàtica de la rotació d'un disc desequilibrat

Observem que per a mantenir la trajectòria circular cal una acceleració normal (**An**) cap al centre de rotació. És a dir, cal una força que "estiri" aquesta massa cap al centre.

Pel principi Newtonià d'acció i reacció, el centre **O** es veurà "estirat" cap a la massa en desequilibri. És a dir, el disc rep una força (**Fc**) aplicada al centre de gir que li provoca una acceleració en direcció a la massa **M**.

Aquesta força va canviant de direcció a mesura que la massa **M** es desplaça per la trajectòria circular. El resultat és que tot el conjunt (en el nostre cas, el ventilador) es va desplaçant cap al centre de masses que al seu temps està en rotació. Per tant, el ventilador en conjunt també descriu un moviment circular, com si orbités respecte al seu eix de gir. Aquest moviment és el que origina les vibracions que observem quan un ventilador està mal equilibrat.



## 1. Suposicions per simplificar el model teòric

Per simplificar el model teòric, farem una sèrie de **suposicions** que ens introduiran errors els quals intentarem corregir posteriorment:

- El ventilador gira sobre l'eix gravitatori de manera que no sofreix inclinacions que per gravetat aportin forces en sentit horitzontal. Tampoc considerarem les forces per l'efecte giroscòpic al suposar que no canvia l'angle de gir.
- El ventilador gira lliurement, sense forces de fricció ni resistències elàstiques produïdes pels ancoratges.

Començarem mesurant les acceleracions del conjunt giratori amb un acceleròmetre. A partir d'aquestes acceleracions podem aplicar la segona llei de Newton ( $F = m \cdot a$ ) per calcular el valor de la força centrífuga a neutralitzar ( $F_c$ ).

## 2. Determinació de la "massa equivalent" del conjunt

Veiem que per aplicar la segona Llei de Newton caldria conèixer la massa del ventilador. Una opció seria desenganxar-lo del sostre i pesar-lo amb un dinamòmetre, fet que d'una banda faria més laboriosa l'operació d'equilibrat i d'altra banda, aquesta dada no ens aportaria informació sobre les resistències elàstiques, friccions o altres forces que hem negligit en les suposicions inicials.

L'opció que considerem més idònia és calcular una dada que en direm "massa equivalent" del ventilador. Per a això, col·locarem una pes de prova conegut a una distància concreta de l'eix de gir, posarem el ventilador en funcionament i analitzarem el canvi produït en les acceleracions respecte a les vibracions inicials. Conegudes la velocitat de rotació, la massa del pes prova i la distància a l'eix de gir, sabrem la força efectuada per aquest pes de prova. Observant els canvis en les acceleracions del sistema, podem calcular la "massa equivalent" del ventilador, és a dir, la oposició del sistema al canvi de velocitat.

Aquesta massa equivalent conté la informació sobre les reaccions del ventilador degudes a friccions, elasticitat del ancoratges, inclinacions i altres valors inicialment negligits. És evident però que el comportament de tots aquests factors no serà lineal. És per això que cal aproximar la massa equivalent als valors de les acceleracions corresponents a la vibració que volem corregir. Així doncs, seguint els criteris de la bibliografia, el pes de prova ens haurà de donar una variació de entre el 30 i el 100% de l'acceleració respecte a l'acceleració inicialment mesurada al ventilador.

### 3. Correcció del desequilibri amb contrapesos

Per generar una força igual i de sentit contrari, posarem un contrapès ( $m_1$ ) de massa coneguda com s'indica a la figura 11. Com no podem situar el contrapès en qualsevol angle del disc de rotació, cercarem el mateix efecte situant 2 contrapesos en les aspes més properes a la direcció de  $m_1$ , de manera que la suma vectorial de les forces exercides per aquests contrapesos es correspongui amb la força desitjada en la direcció  $m_1$ .

Els contrapesos giraran a la velocitat angular del ventilador, la qual podem determinar amb un sensor òptic que detecti el pas d'una pala. Així doncs, aquests contrapesos sofriran una acceleració normal, "estirant" per reacció el centre de gir en direcció contrària a la força realitzada per la massa  $M$ .

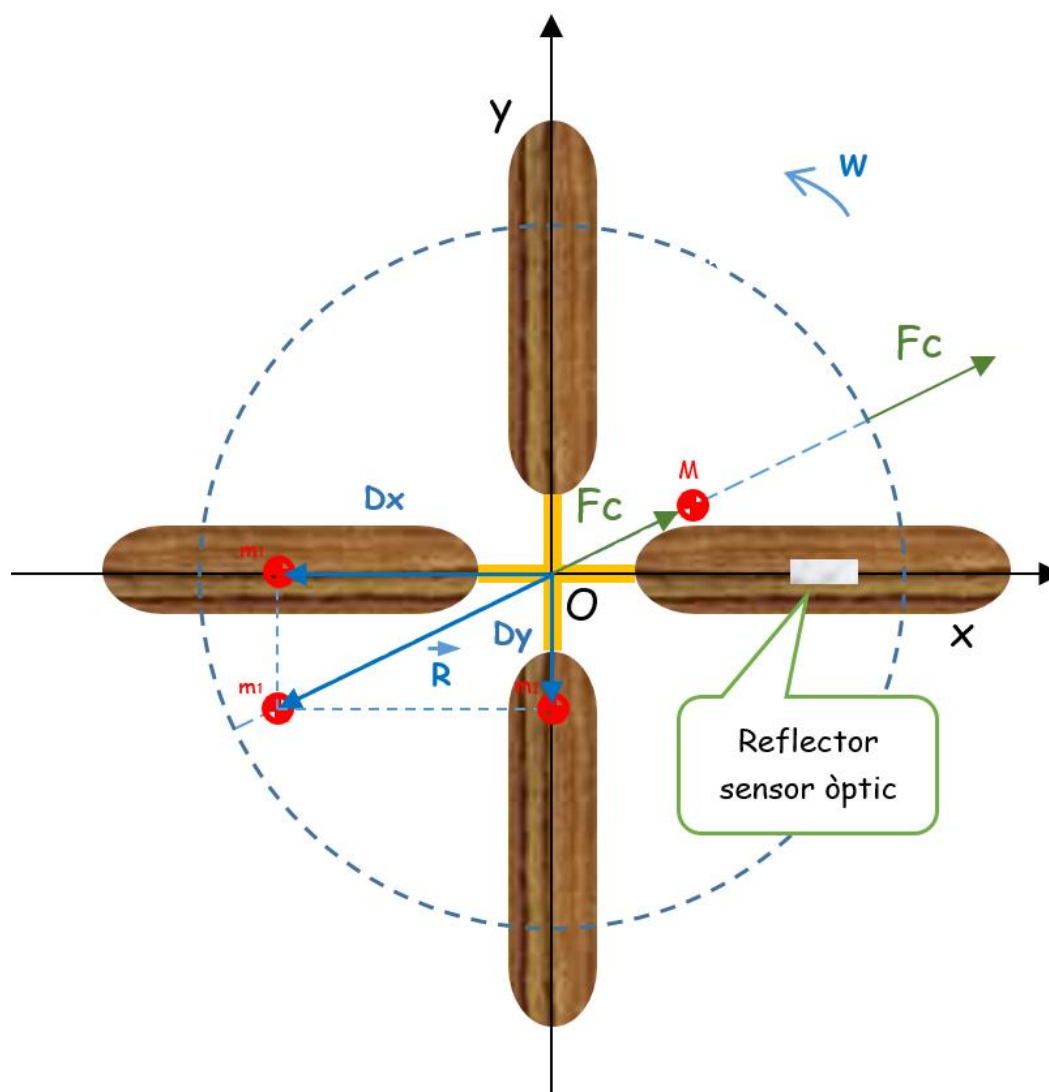


Figura 11: Posició de les masses  $m_1$  per neutralitzar la força  $F_c$

En realitat el que estem fent és reassignar el centre de masses amb una nova distribució sobre el conjunt giratori. Ho fem aplicant les equacions de la dinàmica i la cinemàtica. Per saber on posar els contrapesos. Igualarem la força centrífuga ( $F_c$ ) que obtenim mesurant les acceleracions aplicades a una massa equivalent, a una altra força igual i de sentit contrari ( $-F_c$ ) que calculem com el producte de la massa del contrapès ( $m_1$ ) amb l'acceleració normal d'aquest ( $An(m_1)$ ).

La acceleració normal causada per la massa del contrapès " $An(m_1)$ " és en funció de la velocitat angular ( $\omega$ ) i de la distància al centre de rotació, indicada sovint com a radi de rotació ( $R$ ). En aquest cas s'aplica l'equació  $An(m_1) = R \cdot \omega^2$ .

Així doncs partim del sistema d'equacions [1]:

$$\begin{aligned} F_c &= M \cdot An(M) \\ -F_c &= m_1 \cdot An(m_1) = m_1 \cdot R \cdot \omega^2 \end{aligned} \quad [1]$$

On:

$M$  és la massa del ventilador.

$An(M)$  és l'acceleració que mesurem amb l'acceleròmetre.

$M$  és la massa equivalent del ventildor

$m_1$  és la massa del contrapès

$An(m_1)$  és l'acceleració normal que té el contrapès  $m_1$

$R$  és la distància al centre de la massa  $m_1$

$\omega$  és la velocitat angular a la que gira el ventilador

Totes les variables del sistema d'equacions [1] són conegudes excepte  $m_1$  i  $R$ . Aquestes les podem triar indistintament mentre compleixin l'equació [2]:

$$M \cdot An(M) = m_1 \cdot R \cdot \omega^2 \quad [2]$$

Pel que fa a la distància del contrapès al centre de rotació  $R$ , hem de tenir en compte que si posem els contrapesos sobre les aspes del ventilador,  $R$  sempre serà més petit que la longitud de l'aspa. Per tant haurem de tenir en compte aquest paràmetre del ventilador com a màxim pel valor de  $R$ .

Pel que fa a la massa  $m_1$ , serà una massa triada per nosaltres. En aquest cas usarem monedes perquè tenen un pes estandarditzat i conegut. A més facilitaran la disponibilitat de pesos per a un operari que vulgui aplicar aquest sistema. Usarem 7 pesos entre 2,3 i 8,5grams. Si cap dels pesos és suficient per corregir el desequilibri el programa ens indicarà error. A la figura 12 s'indiquen els pesos d'algunes monedes de la zona Euro.



	<b>1 céntimo</b> diámetro: 16,25 mm espesor: 1,67 mm peso: 2,30 g
	<b>2 céntimos</b> diámetro: 18,75 mm espesor: 1,67 mm peso: 3,06 g
	<b>5 céntimos</b> diámetro: 21,25 mm espesor: 1,67 mm peso: 3,92 g
	<b>10 céntimos</b> diámetro: 19,75 mm espesor: 1,93 mm peso: 4,10 g
	<b>20 céntimos</b> diámetro: 22,25 mm espesor: 2,14 mm peso: 5,74 g

Figura 12: Combinacions de monedes que usarem de contrapesos

El programa comprovarà si amb la massa del primer pes col·locada en un radi  $R$  igual a la longitud de l'aspa és suficient. En cas contrari, agafarà el següent pes, i així successivament fins a trobar un pes suficient o donar error.

Un cop definit el pes ( $m_1$ ) que usarem, ja podem calcular el valor de  $R$  amb el sistema d'equacions [1].

Com hem dit, l'acceleració va girant a mesura que ho fa el ventilador. La dada d'acceleració sobre la que farem els càlculs serà el seu mòdul i angle en el moment en què l'aspa que té el reflector (figura 11) passa per davant del sensor òptic. Com a referència considerarem aquesta aspa situada sobre la part positiva de l'eix de les  $X$ , així situarem les 4 aspes sobre els eixos  $X$  i  $Y$ .

## B. Materials

En primer lloc hem de seleccionar els transductors que ens aportaran les dades que necessitem sobre el comportament del ventilador. Per analitzar moviments podem usar transductors de posició, velocitat o acceleració. Com hem vist al fonament teòric, ens basarem amb les acceleracions que provoquen les vibracions. Avui dia disposem d'acceleròmetres amb electrònica integrada molt assequibles i precisos.

Per relacionar les acceleracions al llarg del temps amb la posició de les aspes usarem un sensor que detecti el pas d'una aspa. Així podrem determinar l'angle del vector acceleració respecte a l'aspa que detecta el sensor. Per emmagatzemar, tractar i transferir les dades a l'ordinador usarem un microcontrolador.

### 1. Sensor de rotació

Elegim un transductor òptic perquè és el que crea menys afectació sobre l'element giratori. Un sensor magnètic, d'inducció o de contactes aportaria masses o friccions que podrien alterar els resultats. Entre els sensors òptics en triem un de baix cost, doncs és suficient per la finalitat que perseguim.

Partirem del transductor òptic JMN SENSOR CNY7 (figura 13), el qual està format per un led emissor d'infrarojos i un transistor de base oberta que permetrà el pas de corrent en funció dels infrarojos que li siguin reflectits. En una aspa del ventilador hi posarem un reflectant que exciti el sensor.

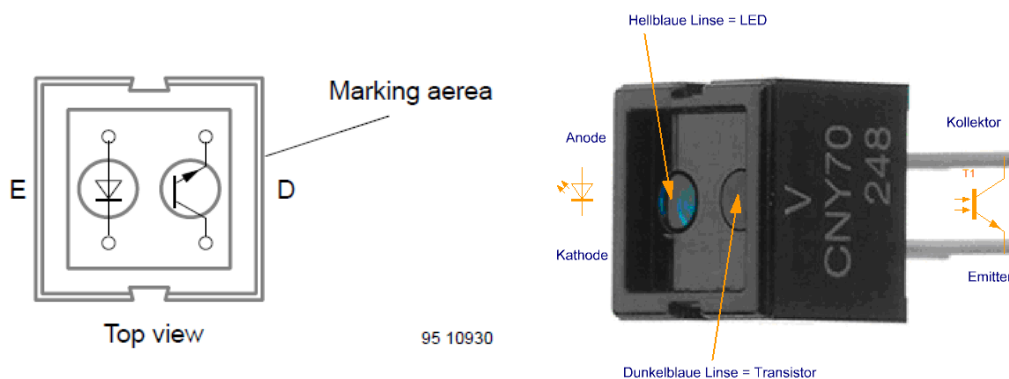
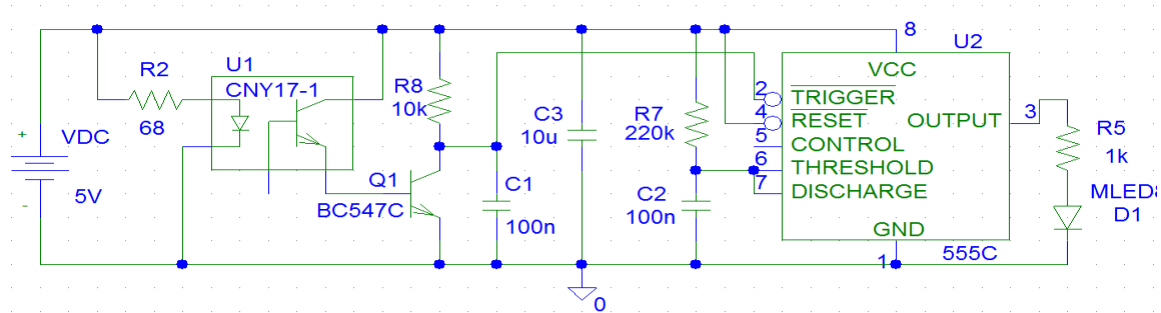


Figura 13: Sensor òptic JMN SENSOR CNY7

Per tal de convertir la senyal analògica d'intensitat de corrent que passa pel transductor òptic en un senyal digital en tensió, caldrà condicionar la senyal amb el circuit que s'indica a la figura 14. Primerament amplifiquem la senyal amb un transistor per tal de millorar la sensibilitat. Després l'aplicarem a un circuit monostable que ens evitarà rebots (captar més d'un flanc per un pas d'aspa) i ens donarà una sortida digital en tensió. El monostable dona polsos d'uns 32 ms més el temps que el sensor estigui detectant l'aspa. Això ens permet una velocitat de lectura de fins a 30 voltes per segon, mentre que el ventilador que usarem per les proves no arriba a les 5 voltes per segon.



**Figura 14: Esquema electrònic del sensor òptic**

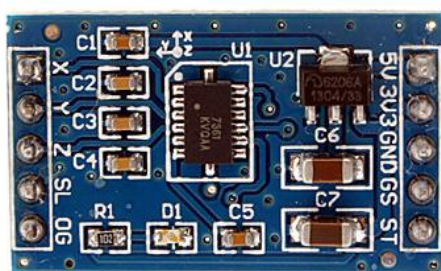
A la figura 15 es pot veure el sensor òptic muntat. A la Taula 1: Connexions al microcontrolador Taula 1 s'indiquen les connexions del sensor al microcontrolador.



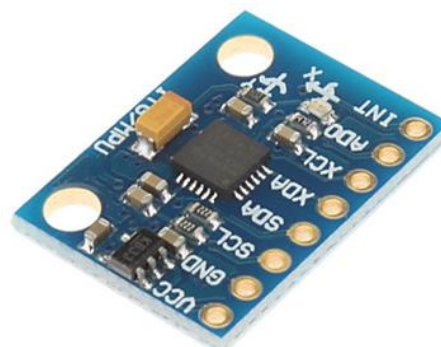
### Figura 15: Sensor òptic

## 2. Sensor acceleròmetre

Hem provat 2 tipus de sensors. Un d'analògic de 3 eixos, model MMA7361 (Figura 16.1) i un de digital de 3 eixos, model GY-521, que amb el microxip MPU6050 incorpora un giroscopi i un termòmetre. A més suporta comunicació I2C (Figura 16.2).



**Figura 16.1: Acceleròmetre analògic**



**Figura 17.2: Acceleròmetre digital**

En un primer prototip vam utilitzar el sensor analògic amb una sensibilitat de 800mV/g (on g és la acceleració de la gravetat), és a dir, 8 vegades superior al mínim recomanat per la bibliografia. El vam connectar amb un microcontrolador Arduino Uno, però aquest acceleròmetre presentava inconvenients respecte al acceleròmetre digital. El temps de lectura era més elevat, doncs el microcontrolador havia de fer la conversió analògica digital. La precisió molt més baixa, doncs l'Arduino Uno converteix les entrades analògiques entre 0 i 5 volts a paraules de 10 bits. Com que l'acceleròmetre dóna una sortida entre 0.1 i 3.2 volts, ja perdíem el 38% dels valors possibles amb 10 bits. És a dir, acabàvem disposant de 634 valors per a un rang de +/- 1,5g. La resolució final quedava limitada a 4.7 mili g.

D'altra banda calia considerar errors de lectura per interferències, doncs les entrades analògiques en són molt sensibles. En canvi, l'acceleròmetre digital dóna una sortida de 16 bits per a un rang de +/- 2g. És a dir, una resolució de 0.06 mili g, 78 cops superior a l'analògic. La lectura també és molt més ràpida que en l'acceleròmetre analògic, podent arribar a una freqüència de mostreig de 1kHz, i alhora és menys sensible a interferències i té una resposta molt més lineal.

Amb la equació [3] es fa la conversió del valor digital de 16 bits per a +/- 2g, a les unitats d'acceleració del sistema internacional [m/s<sup>2</sup>].

$$\text{Acceleració} = 2 \cdot 9.81 \cdot [\text{valor digital d'acceleració en 16 bits}] / 2^{15}; \quad [3]$$

### 3. Microcontrolador

Vam provar els microcontroladors de la figura 18. Finalment vam optar pel STM32F4 Discovery per disposar de majors prestacions en velocitat de càlcul, gestió d'interrupcions i sobretot disponibilitat de llibreries a l'hora d'implementar la comunicació I2C amb l'acceleròmetre digital. A més, el microcontrolador STM32F4 també es distribueix amb una placa que incorpora una pantalla tàctil, amb la qual es podria donar continuïtat al projecte aprofitant el software que utilitzarem pel treball.

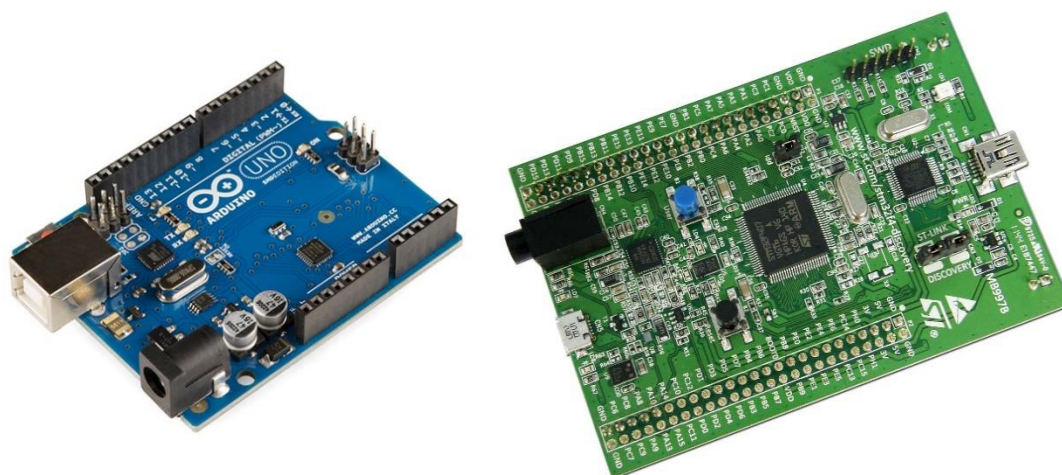


Figura 18: Microcontroladors: (esq.) Arduino Uno i (dr.) STM32F4 Discovery



#### 4. Connexions del conjunt

A més del material descrit anteriorment, necessitarem un connector USB a RS232 per comunicar el microcontrolador i l'ordinador. El conjunt connectat es mostra a la figura 19. A la Taula 1 s'indiquen les connexions als pins del microcontrolador.

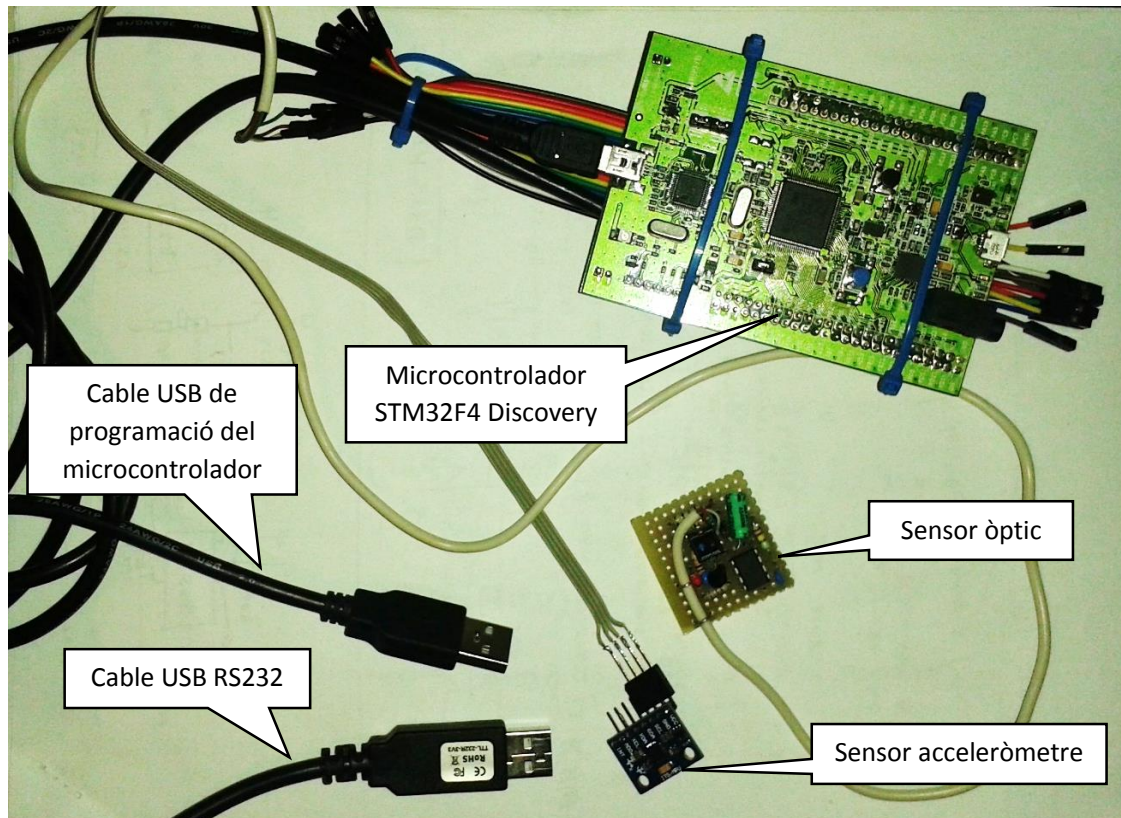


Figura 19: Hardware necessari per a l'equilibrat de ventiladors

Taula 1: Connexions al microcontrolador ( $\mu C$ )

Pins $\mu C$	Cable sensor òptic	Cable acceleròmetre	Cable USB-RS232
Vcc (5V)	Vermell	Vermell	Vermell No connecta
Vcc (3.3V)			
Ground (0V)	Negre	Marró	Negre Pont amb AD0
PA1	Verd		
PA2 (USART RX)			Blanc
PA3 (USART TX)			Verd
PA8 (I2C_3 SCL)		Taronja	
PC9 (I2C_3 SDA)		Groc	

## C. Desenvolupament del procediment d'equilibrat

### 1. Elaboració del software del microcontrolador i de Matlab.

D'una banda cal realitzar el software del microcontrolador per a què llegeixi les dades del acceleròmetre i del sensor òptic i les envii pel port sèrie a l'ordinador. S'adjunta el programa realitzat amb l'entorn de programació Atollic True Studio a l'annex I.

El programa del microcontrolador s'ha estructurat en estats per adaptar-se millor a un procés industrial amb diferents etapes. A la figura 20 s'indica com cal interpretar el diagrama de flux. A la figura 21 podem observar el diagrama de flux del programa inicial del microcontrolador.

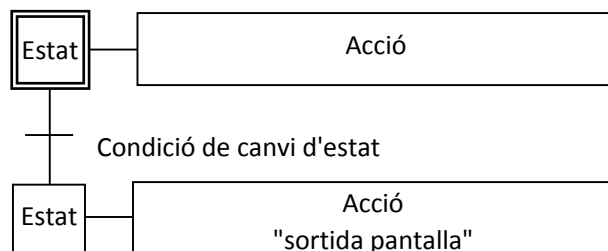


Figura 20: Interpretació del diagrama de flux

Utilitzarem els leds de la placa del microcontrolador per tenir informació. El led vermell canviarà d'estat cada cop que el sensor òptic detecti un pas de l'aspa amb el reflectant. Amb això podrem comprovar si les lectures són correctes o ajustar el sensor si és necessari. El led verd canvia d'estat a la freqüència de mostreig. Així indica que s'està mostrejant la senyal. El led blau ens indicarà fi del programa o error en algun dels estats. Hem creat una funció "alarma(estat)" que el fa parpellejar el nombre de cops corresponent a l'estat en el que s'ha produït l'error, i després d'una pausa de 3 segons, repeteix el parpelleig. També hem configurat un "mode prova" que permet saltar alguns estats de comprovació per agilitzar la presa de dades. El led taronja encès indicarà que hem activat aquest mode.

Així doncs disposem d'un interfase molt simple, però directe entre l'usuari i el microcontrolador. Encara que no puguem visualitzar dades pel port sèrie, tindrem informació d'interès del microcontrolador. Podrem saber l'estat en què s'ha produït un error, si el sensor òptic detecta correctament, si la comunicació entre l'acceleròmetre i el µC es correcta, etc. A la taula 2 es mostra el comportament de la interfase.

Taula 2: interfase  $\mu$ C - usuari

Interfase $\mu$ C	Acció	Significat
Led vermell	Canvi	Detecció pas d'aspa
Led verd	Encès atenuat	Mostreig de senyal
Led taronja	Encès	Mode prova
Led blau	Parpelleig	Error estat núm. parpellejos
Botó blau	Prémer	Continuar

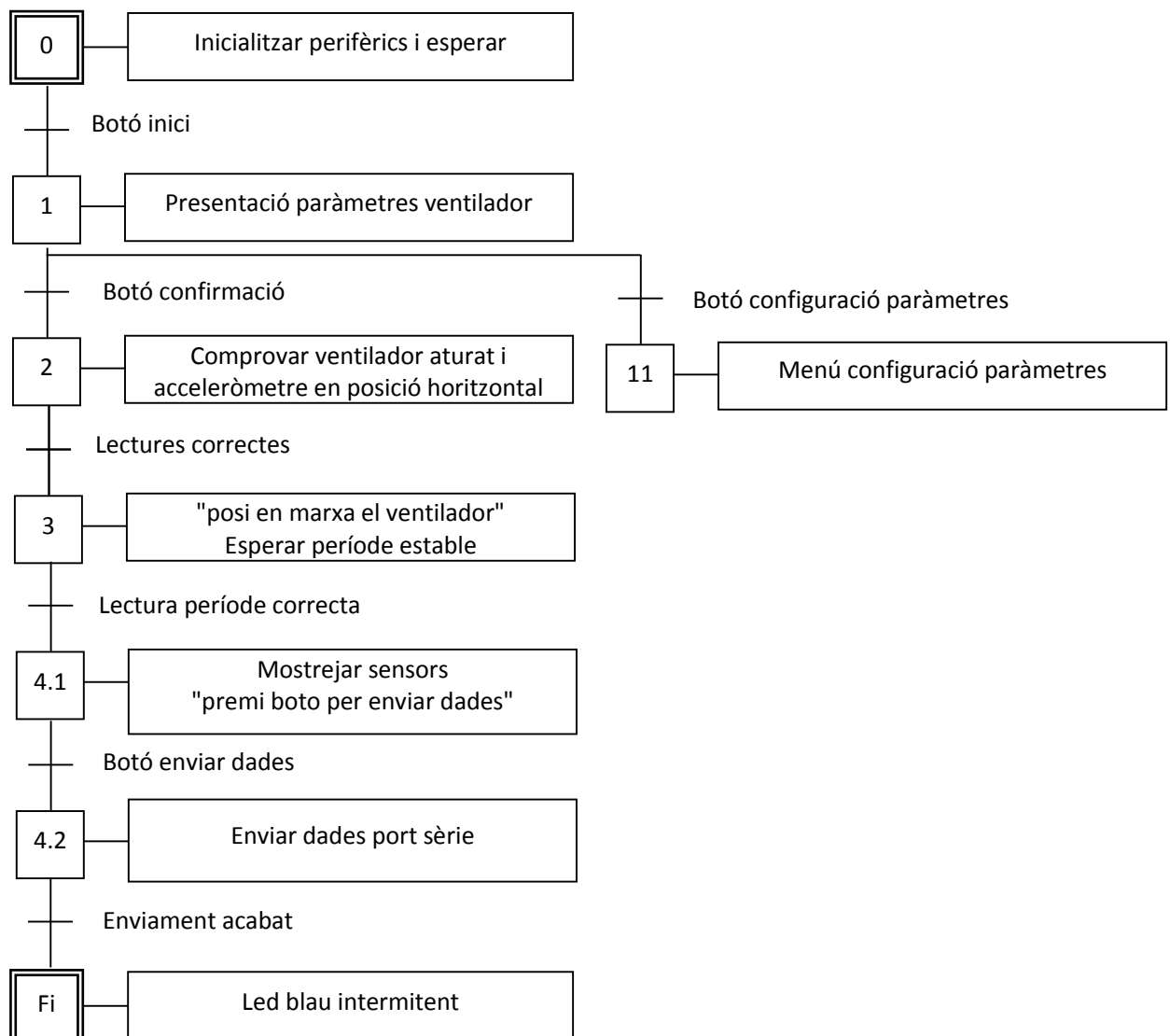


Figura 21: Diagrama de flux del programa inicial del microcontrolador

El software del microcontrolador enviarà les dades al port sèrie per a ser llegides com una matriu, on les files són el número de la mostra i les columnes corresponen a les diferents senyals en l'ordre següent: acceleració eix X, acceleració eix Y, acceleració eix Z i detecció de pas d'aspa.

El microcontrolador pot operar a diferents freqüències de mostreig ( $F_s$ ) però si aquesta és elevada, el port sèrie no es prou ràpid per traspasar les dades en temps real. Així doncs s'ha optat per fer el mostreig i emmagatzemar-lo en la memòria del microcontrolador ( $\mu C$ ) i després enviar les dades al port sèrie a una velocitat de transmissió que no ens saturi el buffer.

D'altra banda cal realitzar el software per a la lectura de dades amb Matlab. Hem creat la funció "Read\_Serial(mostres,  $F_s$ )" per a llegir les dades de la senyal des del port sèrie i graficar-les; la funció "Save\_File(matriu dades, nom arxiu)" per guardar les dades en fitxers; la funció "analisi\_senyal(nom\_arxiu, mostres,  $F_s$ , filtrat, voltes) per graficar els espectres de potència a diferents freqüències entre altres funcions que més endavant descriurem. S'adjunten a l'annex II les diferents funcions creades amb Matlab.

## 2. Captura i emmagatzematge de les dades

L'acceleròmetre es situa a la part central del ventilador mitjançant un imant. És important que quedi ben horitzontal. Aquest tipus de sensor detecta la gravetat. Si està inclinat, els sensors dels eixos X i Y captaran una part de l'acceleració de la gravetat, modificant en un valor constant la senyal capturada al llarg de la lectura. Si la inclinació és petita ho podem admetre, doncs al filtrar la senyal amb un filtre passabanda anul·larem la component contínua introduïda per la gravetat. Però cal tenir en compte que la inclinació ens resta sensibilitat. Per això s'ha fet ajustable l'horitzontalitat del sensor amb 2 petits cargols. El software del microcontrolador comprovarà que l'acceleròmetre en repòs no doni valors superiors a 500 (aproximadament  $0,3\text{m/s}^2$ ), el que correspondria a una inclinació de  $1,75^\circ$ . (La inclinació l'obtenim aïllant  $\alpha$  de l'expressió  $9,81\text{m/s}^2 \cdot \sin(\alpha) = 0,3\text{m/s}^2$ ).

Interessa posar l'acceleròmetre el més prop possible del pla de rotació, i en tot cas, és imprescindible que estigui en la mateixa posició durant tota la operació. A més, cal tenir en compte que l'eix X de l'acceleròmetre ha d'assenyalar la posició del sensor òptic com s'indica a la figura 22.



Figura 22: Orientació eixos acceleròmetre

Posició sensor òptic



La lectura de l'acceleròmetre pot tenir desviació per la variació temperatura. Cal que les dos lectures (la inicial i la del pes de prova) es facin amb poca diferència de temps. I si, per acció de les bombetes o el motor del ventilador, la temperatura augmenta significativament en poc temps, és millor deixar-la estabilitzar abans d'iniciar les lectures.

Per garantir que el sensor òptic funciona correctament, primer es comprovarà que detecti el reflector movent les aspes manualment i observant si s'activa el led que incorpora el propi sensor (Figura 23). Després, amb el ventilador en marxa i el microcontrolador alimentat, observarem que la cadència de parpellejos del led vermell del microcontrolador sigui homogènia. Es pot apropar o allunyar el sensor de les aspes per ajustar la sensibilitat.

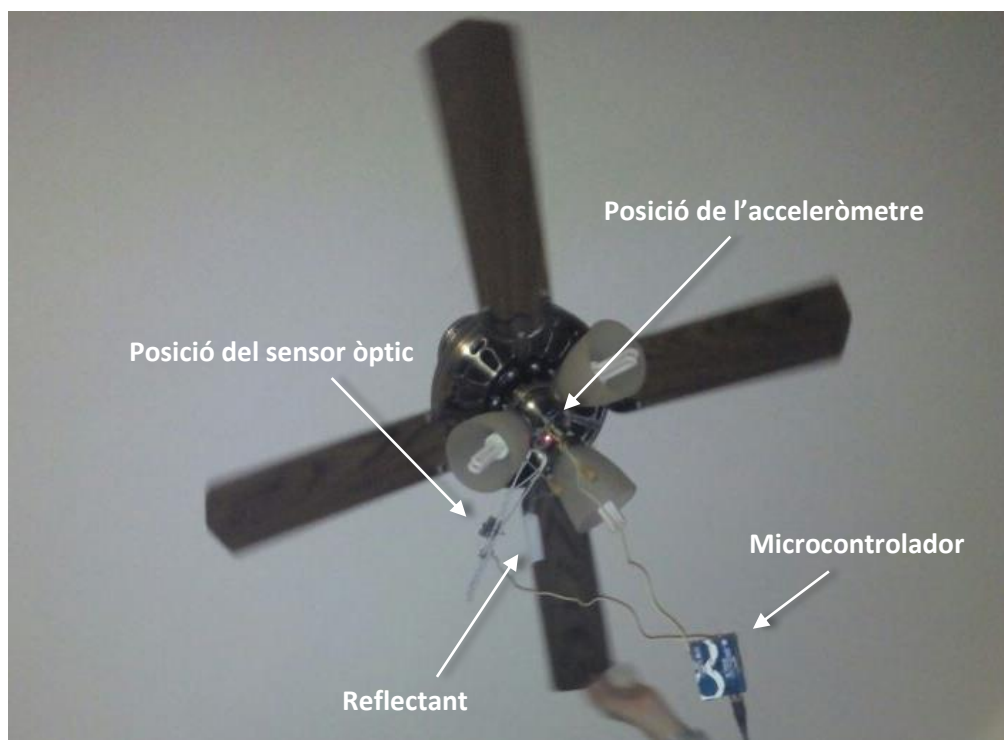
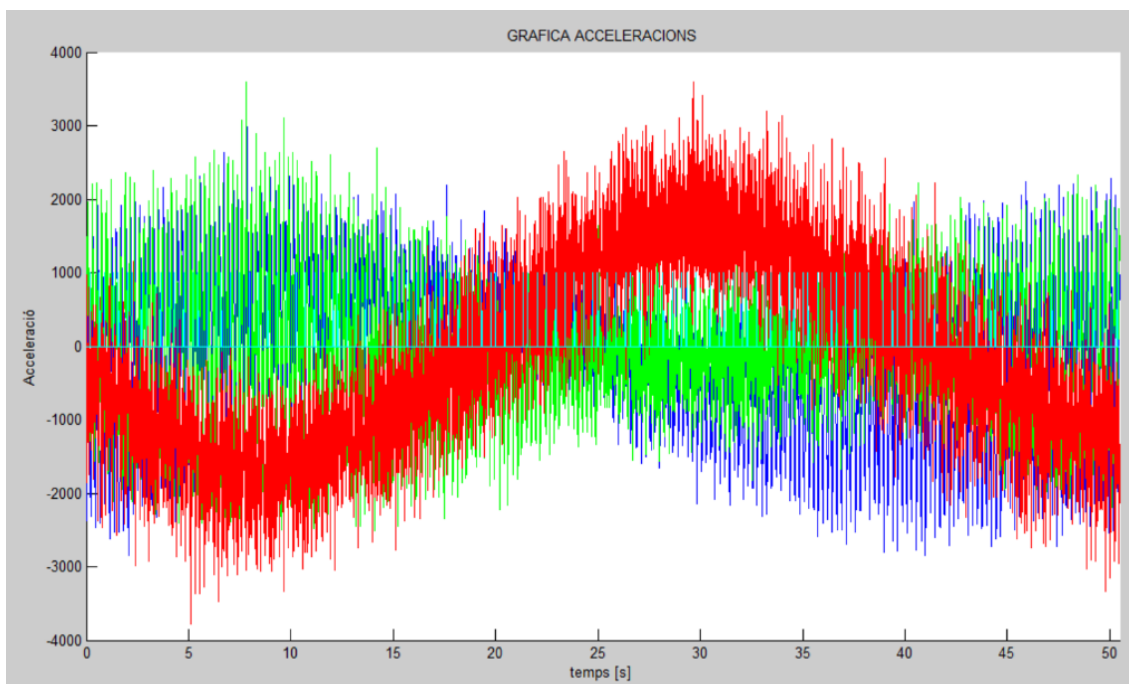


Figura 23: Situació dels components i comprovació del funcionament del sensor òptic

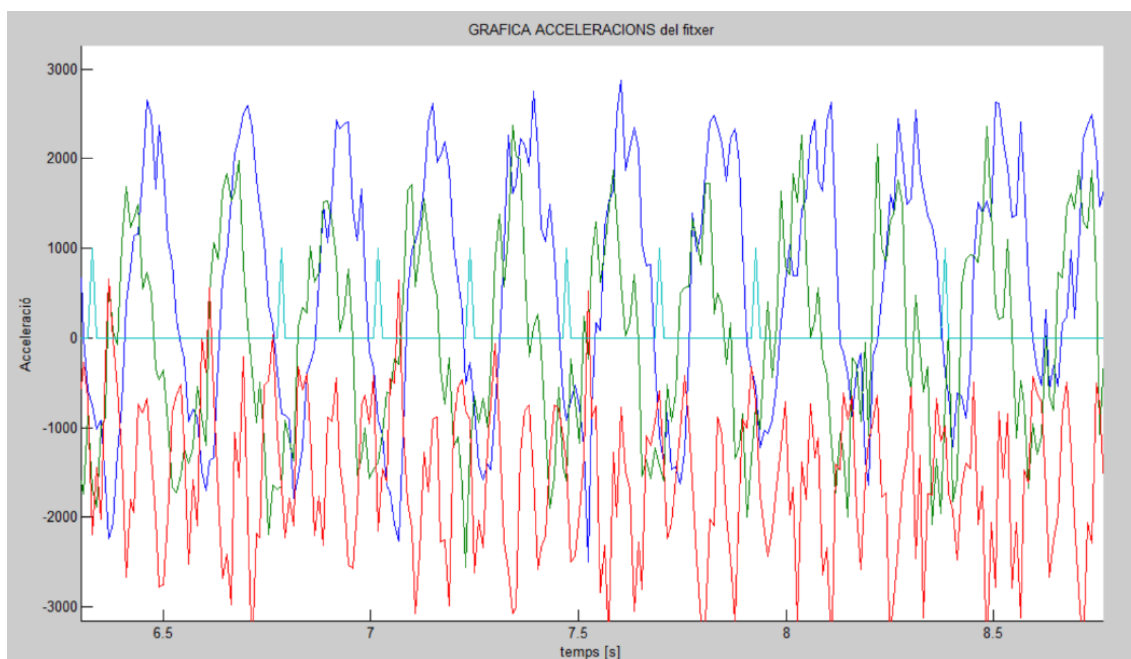
### 3. Anàlisi previ per al tractament de la senyal

Per determinar les freqüències de mostreig ( $F_s$ ) i el tractament que requerirà la senyal cal fer-ne una anàlisi prèvia.

Les gràfiques 1 i 2 presenten el valor d'acceleració que dona el sensor al llarg del temps. En blau l'eix X, en verd, l'eix Y, en vermell l'eix Z i en cian la detecció del pas d'aspa. S'han compensat les lectures inicials de l'acceleròmetre en repòs, de manera que s'omet l'acceleració detectada per la gravetat.



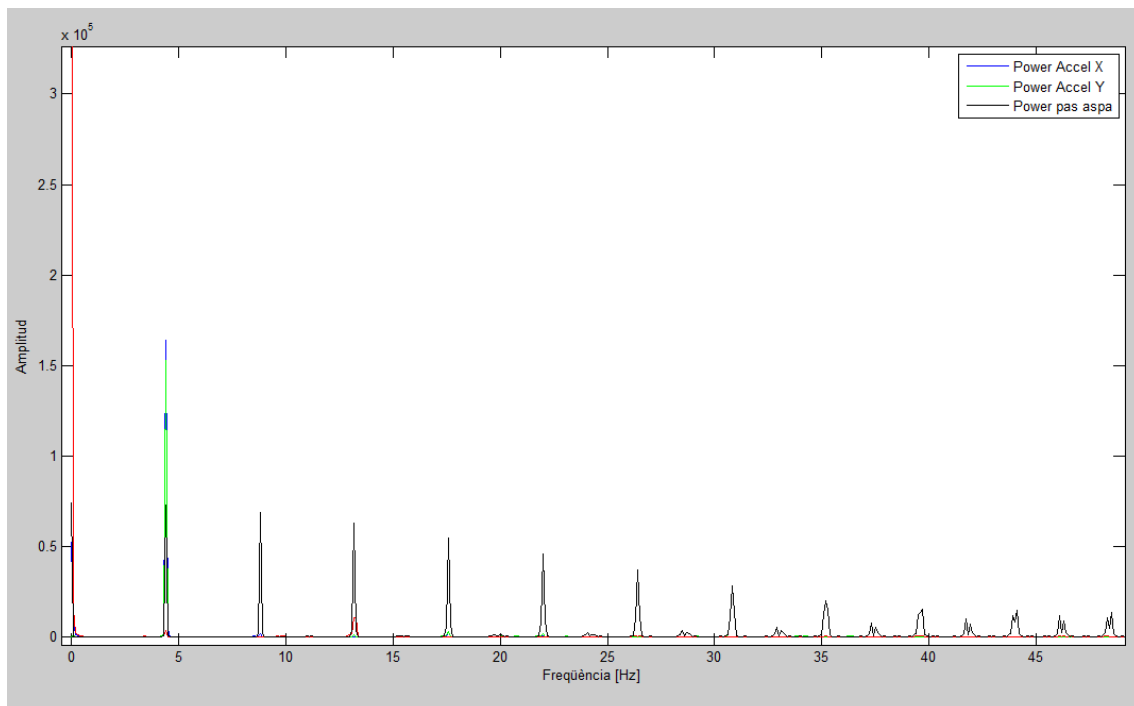
Gràfica 1: (Fs: 99Hz) Senyal captada amb el motor funcionant



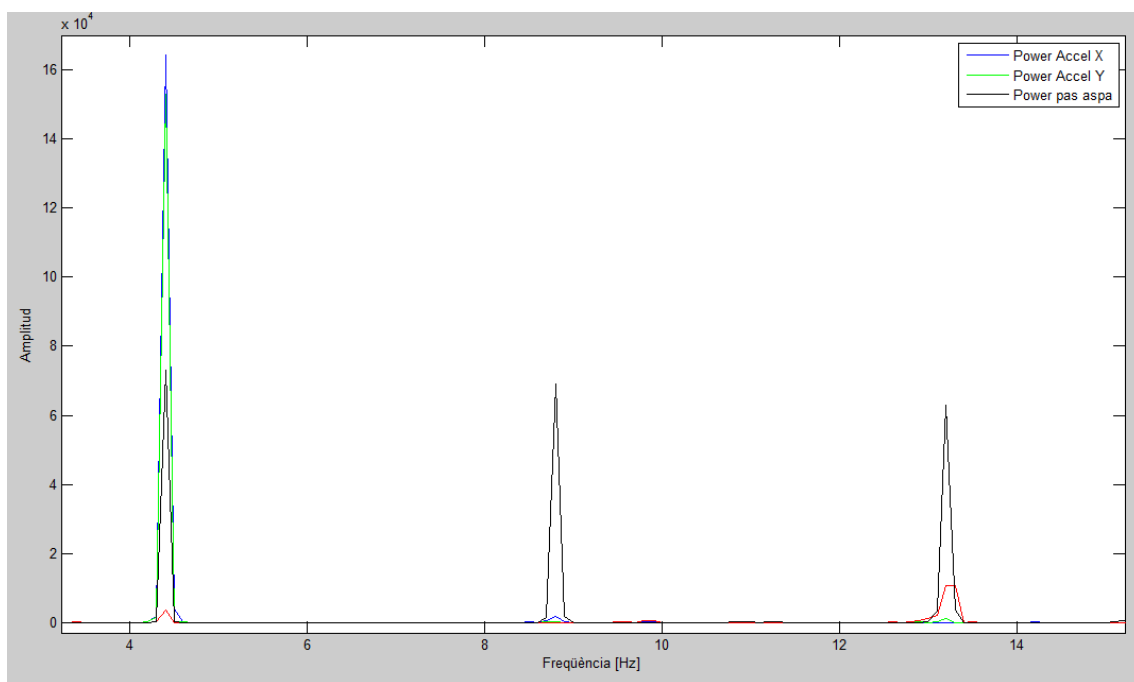
Gràfica 2: (Fs: 99Hz) Ampliació de la gràfica 1

Observem diverses components de freqüència en la senyal. Només una es deu a la rotació de les aspes, per tant caldrà filtrar la senyal amb un filtre passa banda a la freqüència de rotació.

A les gràfiques 3 i 4 analitzem l'espectre de la senyal amb la transformada ràpida de Fourier executada amb Matlab "fft(senyal discreta)".



Gràfica 3: (Fs: 99Hz) Espectre de potencia de la senyal



Gràfica 4: (Fs: 99Hz) Ampliació de l'espectre de potencia gràfica 3

Observem que la freqüència principal de les acceleracions X i Y la tenim a uns 4,4Hz. Aquesta és la freqüència de rotació del ventilador. Les 3 senyals d'acceleració corresponents als 3 eixos tenen component contínua, especialment la Z (paral·lel a la gravetat). Pel que fa a la senyal del pas d'aspa, formada per polsos, té harmònics en tot el rang de freqüències, tal com correspon a uns senyal formada per impulsos.

Curiosament, l'acceleració en Z adquireix una magnitud important a 3 vegades la freqüència de rotació. Això ens indica una oscil·lació en sentit vertical a aquesta freqüència. L'explicació que hem trobat és que les subjeccions del ventilador no tenen la mateixa rigidesa en totes les direccions. A la figura 24 es pot observar que el ventilador es subjecta amb un passador amb certa folgança en una direcció, i amb un cargol que limita els moviments en l'altra.

Fent una comprovació manual confirmem que té més joc (menys rigidesa) en el sentit del passador que en el del cargol. Al vibrar el ventilador per la rotació de les aspes, "xoca" amb els límits superior i inferior de la folgança produint una variació important de l'acceleració vertical. Arriba un moment en què aquests xocs es sincronitzen a 3 cops la velocitat de rotació, de forma anàloga a com una broca que s'encalla en dos punts quan perfora una planxa, acaba dibuixant una mena de triangle ovalat.

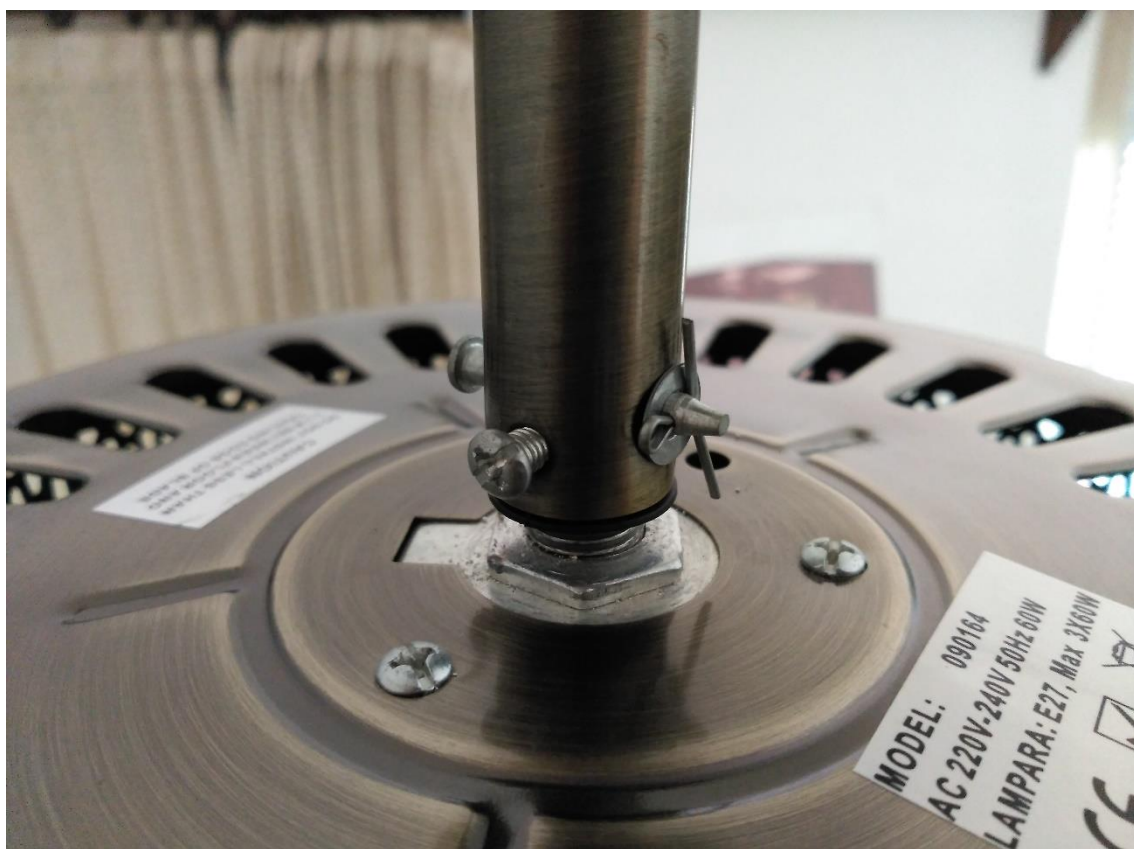
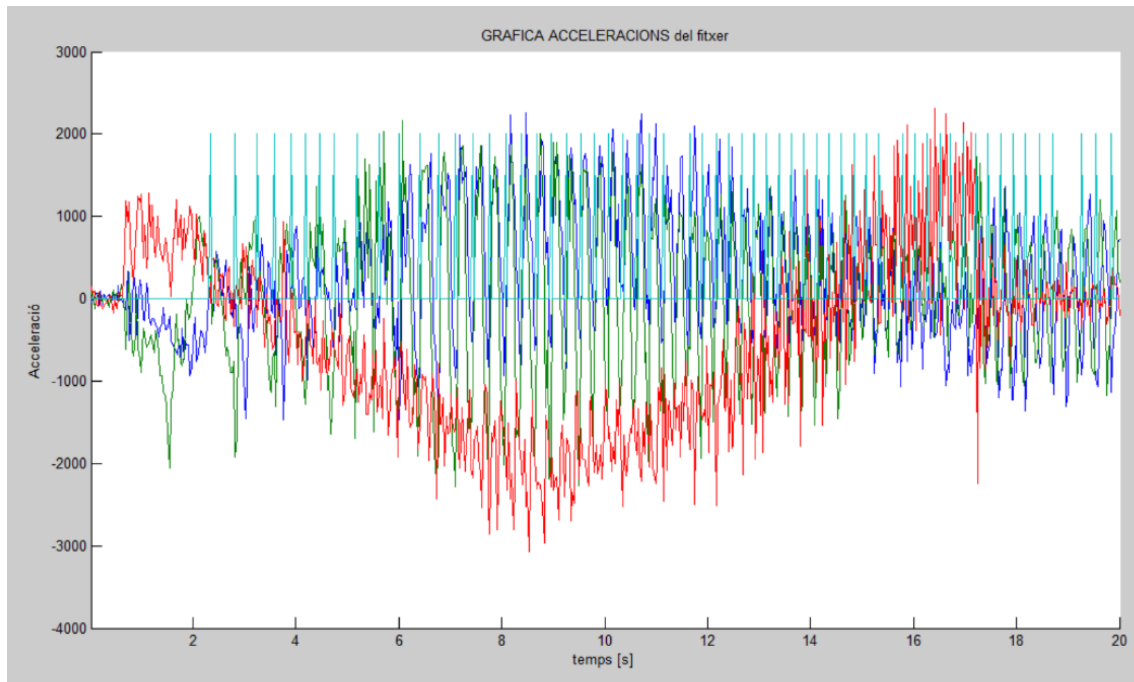


Figura 24: Subjecció del ventilador amb diferent rigidesa segons la direcció

Com veurem més endavant, la diferent rigidesa en una o altre eix ens afectarà a les vibracions i dificultarà l'equilibrat.

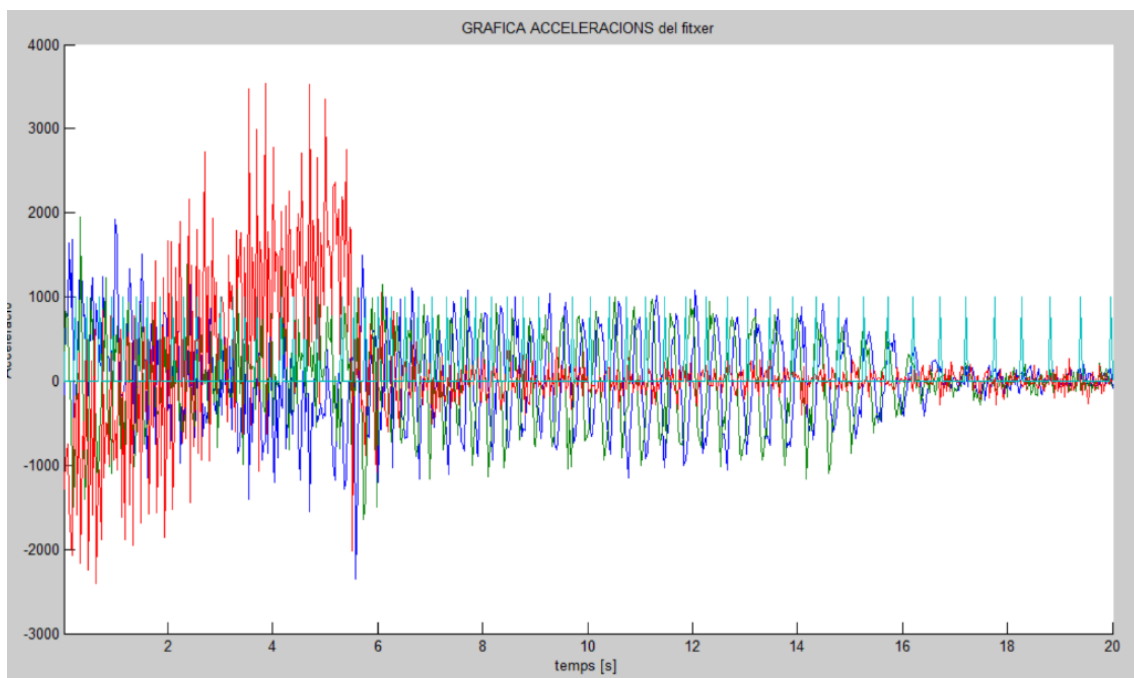
Un altre aspecte de la senyal que cal estudiar és la freqüència que presenta a uns 0,02Hz i que s'observa especialment amb l'oscil·lació de l'acceleració Z en la gràfica 1 al llarg d'un període d'uns 45 segons. Per saber si té relació amb els camps electromagnètics de l'estator i el rotor del ventilador, posem en marxa i aturem el motor mentre prenem mostres com s'observa a la gràfica 5.



Gràfica 5: (Fs: 99Hz) Engegada (segon 1) i aturada (segon 17,2) del motor

Mentre no hi ha moviment tenim valors de pràcticament 0. Al segon 1 es posa en marxa el ventilador i immediatament, abans que les aspes comencin a agafar velocitat i el ventilador vibri, ja observem que la senyal adquireix amplitud. Poc després del segon 17 s'atura motor, i malgrat les vibracions continuen per la rotació inercial de les aspes, les acceleracions deixen de presentar la components de 0.02Hz. Això ens confirma que l'acceleròmetre és sensible als camps electromagnètics del motor.

A la gràfica 6 observem l'efecte anterior i com les acceleracions es van reduint a mesura que les aspes es van aturant.



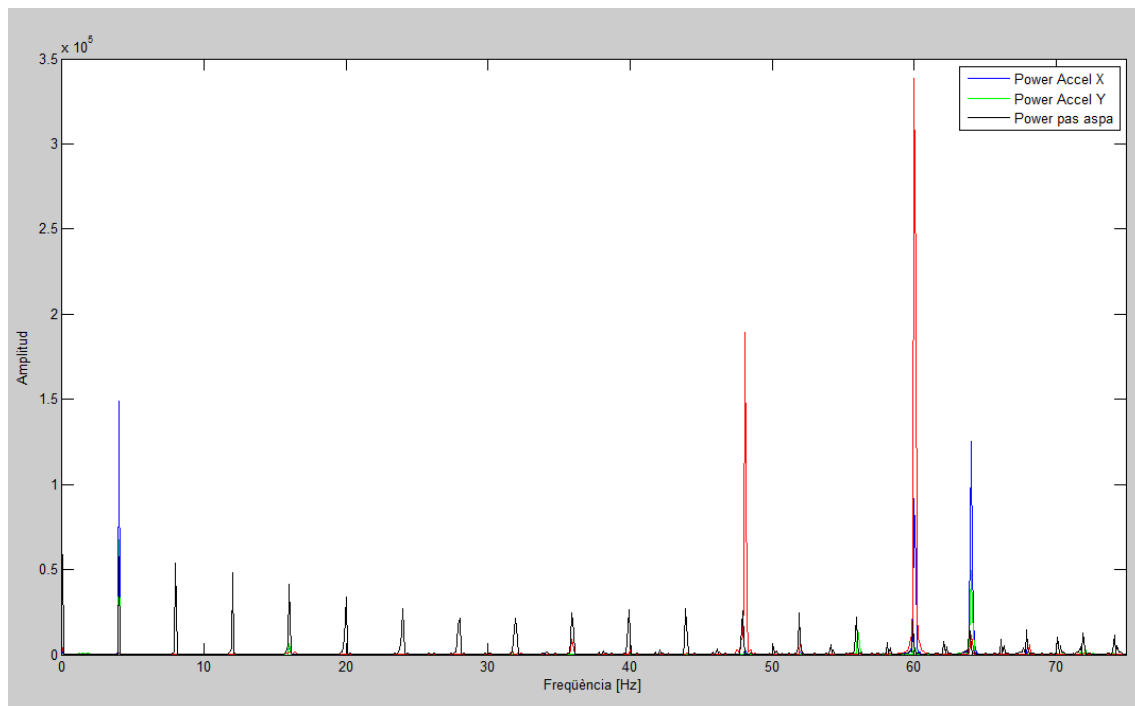
Gràfica 6: (Fs: 33Hz) Aturada del motor i reducció de la velocitat de les aspes

No ens interessen els transitoris d'arrencada i parada, motiu pel qual prendrem les mostres amb la velocitat estabilitzada. Els transitoris només ens interessarien si volguéssim saber la freqüència de ressonància del conjunt. Això seria molt interessant a l'hora de dissenyar el ventilador per tal que la velocitat nominal no coincidís amb la de ressonància. Però en el cas que ens ocupa, només necessitem saber les vibracions quan el ventilador es troba treballant a la velocitat nominal, en situació estacionària.

#### 4. Elecció de la freqüència de mostreig

D'acord amb la bibliografia, la freqüència de mostreig d'una senyal ha de ser com a mínim d'uns 10 cops superior a la freqüència màxima que en vulguem extraure, en el nostre cas uns 5Hz. Hem fet proves amb una  $F_s$  de 33, 66, 99 i 150Hz. Totes aquestes  $F_s$  fan una representació correcta de la senyal. Però com veurem més endavant, ens interessarà trobar el moment dels màxims o els passos per zero de la senyal. Si tenim poques mostres, la mostra que més s'apropi al màxim o al zero pot quedar relativament lluny del instant on aquest màxim o zero es produeix.

Per exemple, amb una freqüència de rotació de 4,5Hz i una  $F_s$  de 99Hz, tindrem 22 mostres per volta, és a dir, a cada mostra el ventilador haurà girat més de  $16^\circ$ . Per millorar la precisió ens cal augmentar la  $F_s$ . Si augmentem la freqüència de mostreig, a 150Hz, observem una component molt important d'acceleracions a 60Hz (gràfica 7) segurament atribuïble als camps electromagnètics i les vibracions del motor. Com que el filtre no eliminarà totalment aquesta freqüència, optem per triar una  $F_s$  de 110Hz que ja elimina totes les freqüències superiors a 55Hz. Serà una freqüència 20 vegades superior a la del ventilador i suficient per la nostre propòsit.



Gràfica 7: ( $F_s$ : 150Hz) Espectre amb forta component de senyal a 60 Hz

## 5. Filtrat

Usarem filtres FIR (Finite Impulse Response) perquè tenen un retard de grup constant i un retard de fase lineal. Hem de tenir en compte que el angle de l'acceleració l'hauréem de calcular a partir dels valors d'acceleració X i Y en un instant concret. Això implica que un cop filtrades les senyals, les mostres d'acceleracions i del pas de l'aspa pel sensor òptic han d'estar sincronitzades (mateix retard de grup) i han de tenir idèntic desfasament (mateix retard de fase).

Un altre avantatge important dels filtres FIR és la seva fàcil implementació en un microcontrolador. En el cas del STM32F4, existeixen llibreries amb la funció de filtrat amb filtres FIR. L'inconvenient que presenten és que per aconseguir el mateix efecte que un filtre IIR (Infinite Impulse Response –amb zeros i pols), els FIR necessiten un ordre molt superior. En les proves realitzades unes 30 vegades superior. Però això no sempre implica major temps de càlcul al microcontrolador, doncs l'algoritme dels filtres IIR és més complex.

Matlab ens aporta l'eina FDATool que permet calcular els coeficients dels filtres de forma molt senzilla. Simplement cal donar les especificacions del filtre (tipus,  $F_s$ , freqüències de tall, atenuació...) i genera el filtre, les característiques i resposta del qual es poden visualitzar amb la mateixa eina (figura 25). A més, permet exportar els coeficients per a incloure'ls en el programa del microcontrolador.

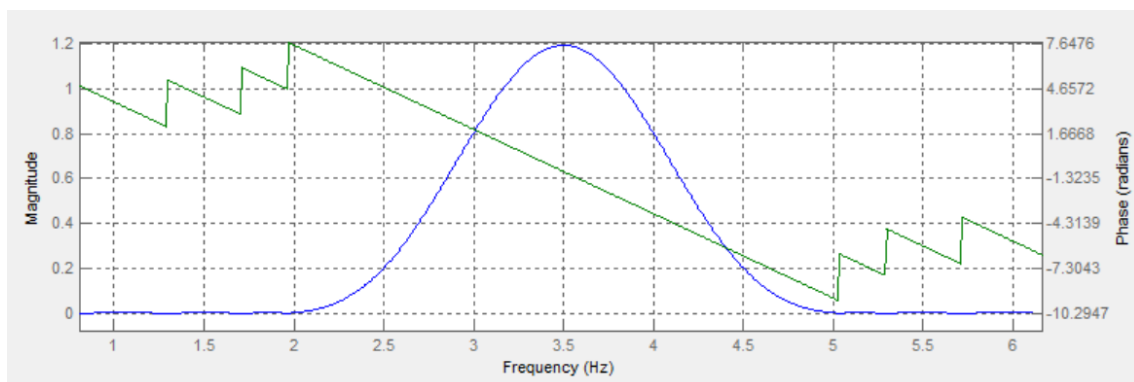
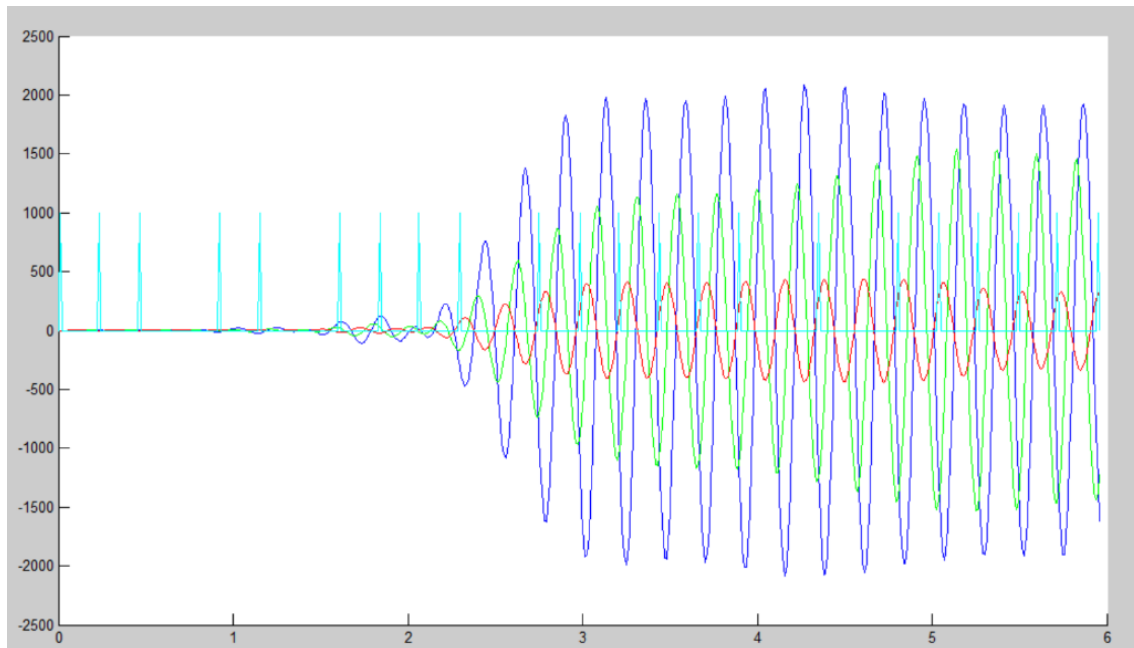


Figura 25: Espectre filtre FIR passa banda 3-4Hz ordre 197,  $F_s$  110hz. Blau: magnitud, verd: fase

La variació de la resposta en amplitud a la banda de pas del filtre no significaran un problema pel càlcul dels contrapesos atès que la freqüència de rotació, amb i sense pes de prova, varia de forma insignificant (de l'ordre de 0,05Hz). Per tant els càlculs del desequilibrat original i el provocat pel pes de prova es realitzen sobre el mateix factor d'amplificació o atenuació.

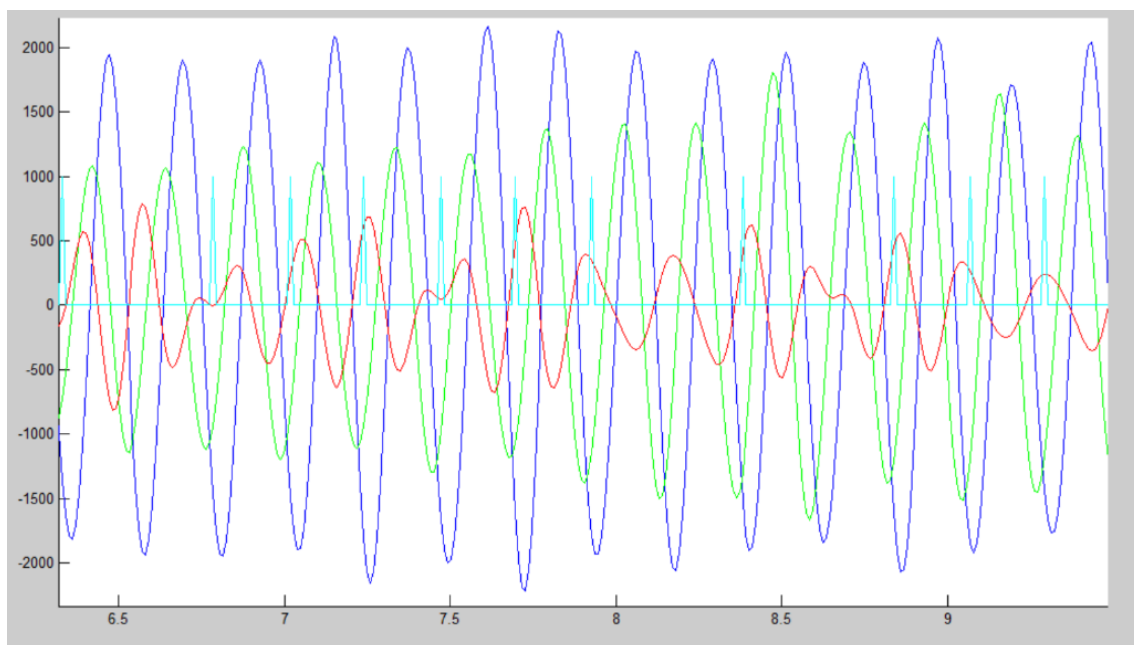
Sabem que a major ordre, el filtre tindrà una resposta més precisa. Hem fet proves amb filtres d'ordre 375 sense que representi cap problema per a la potència de càlcul del microcontrolador. En canvi, si que és un inconvenient el retard de grup del filtre perquè perdem una franja de temps de la senyal que voldrem analitzar. A la gràfica 8 observem l'efecte del retard d'un filtre de 248 coeficients aplicat a una senyal amb  $F_s$  de 99Hz. El retard és directament proporcional al nombre de coeficients del filtre. Així doncs, caldrà usar un filtre d'ordre petit per reduir el retard, però suficient per atenuar les freqüències no desitjades.





Gràfica 8: (Fs: 99Hz) Retard introduït per un filtre de 248 coeficients

Un altre aspecte a decidir és l'ample de banda del filtre. Podem trobar ventiladors que treballin a diferents velocitats o ens pot interessar equilibrar per les diferents marques d'un mateix ventilador. Per tant desconexim la freqüència exacta amb què haurem de treballar, però tot i això podem preveure que anirà dels 2 als 5 Hz. A la gràfica 9 s'observa la senyal filtrada amb un filtre passabanda de 2 a 5Hz). Es detecta que ens entra una freqüència no desitjada, especialment a l'acceleració Z (vermell).

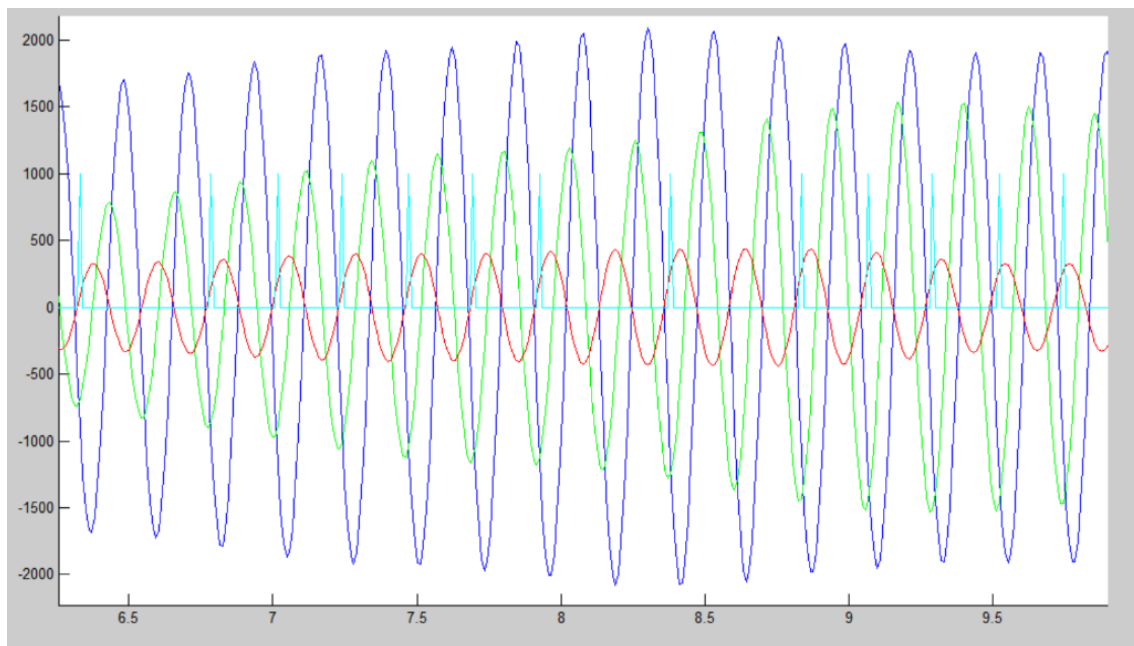


Gràfica 9: (Fs: 110Hz) Senyal filtrada amb passabanda 2-5Hz

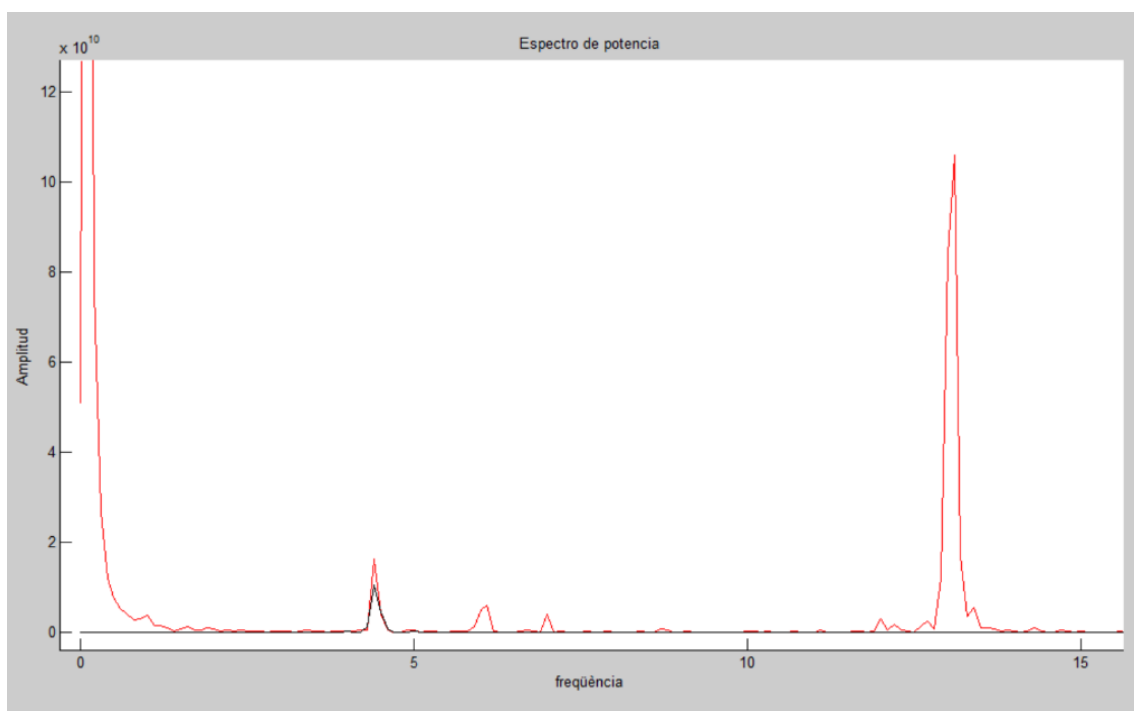
Per corregir aquest problema optem per dissenyar 3 filtres amb una banda de pas d'1Hz. (de 2 a 3, de 3 a 4 i de 4 a 5). El programa del microcontrolador calcularà la freqüència de rotació a partir del període que ens dona el sensor òptic i aplicarà el filtre que



correspongui a la senyal. A la gràfica 10 es presenta la mateixa senyal de la gràfica 9 filtrada amb un passabanda 4-5Hz. A la gràfica 11 en traça negra veiem com queda l'espectre de l'acceleració Z després d'aplicar aquest filtre de banda més estreta.



Gràfica 10: (Fs: 110Hz) Senyal filtrada amb passabanda 4-5Hz



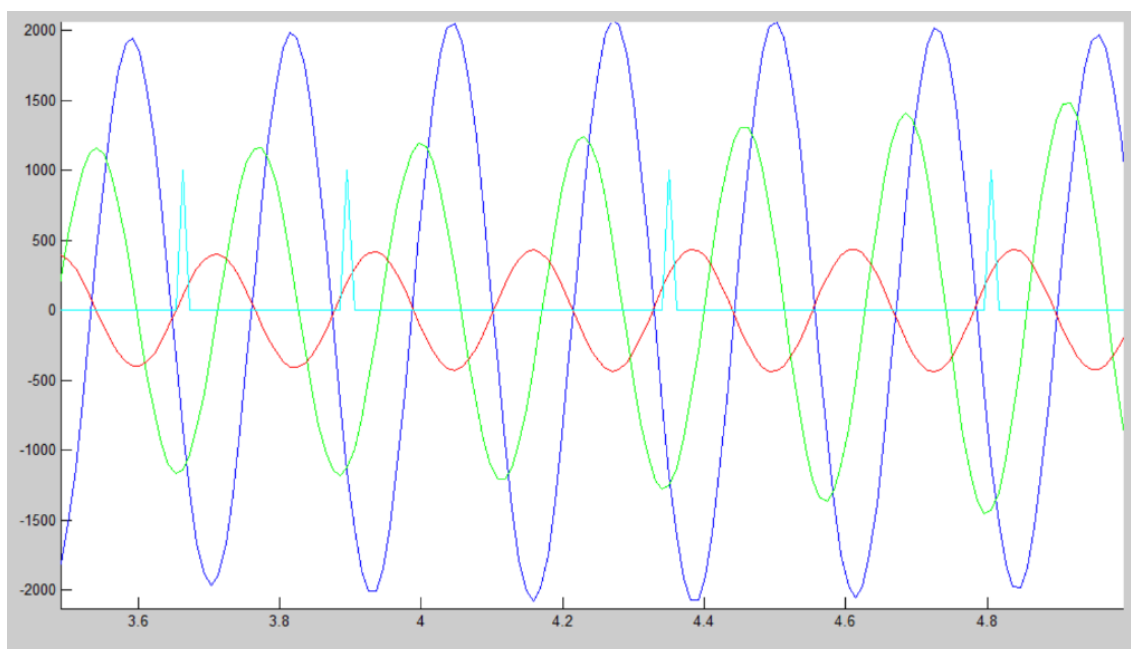
Gràfica 11: (Fs: 110Hz) Acel. Z. En negre l'espectre filtrat amb passabanda 4-5Hz

A l'annex III s'adjunten els coeficients dels 3 filtres utilitzats i la Figura 40 amb el disseny del filtre passabanda de 3 a 4 Hz, d'ordre 197. Els 3 filtres utilitzats tindran les mateixes característiques canviant només 1 Hz les freqüències de tall.

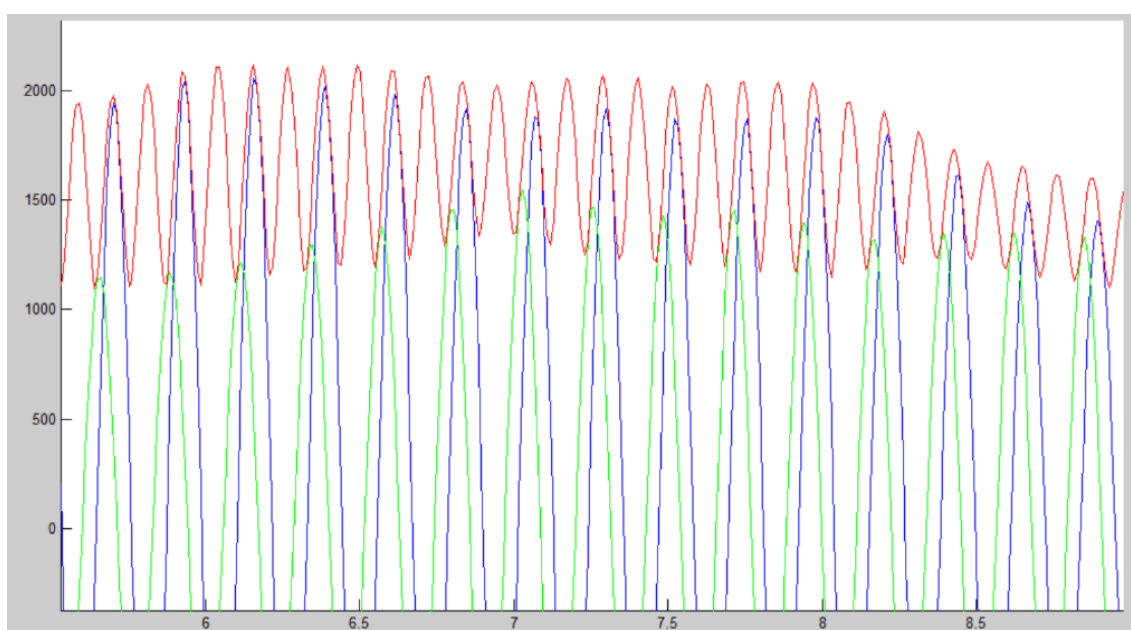
## 6. Monitorització i anàlisi de les dades de les vibracions

D'acord amb el model teòric, si sumem vectorialment les components de l'acceleració X i Y d'acord amb l'equació [5], ens hauria de donar un mòdul d'acceleració constant que va canviant de direcció amb la rotació de les aspes. En canvi, a la gràfica 12 comprovem que no és així. En primer lloc estem negligint l'acceleració en l'eix Z, i en segon lloc, observem que l'acceleració en X (blau) té més amplitud que en Y (verd). A la gràfica 13 s'indica en vermell la suma vectorial de les acceleracions X i Y. El seu valor no es constant ni uniforme al llarg del temps.

$$|Acceleració \text{ en el pla } XY| = \sqrt{Accel. \ x^2 + Accel. \ y^2} \quad [5]$$

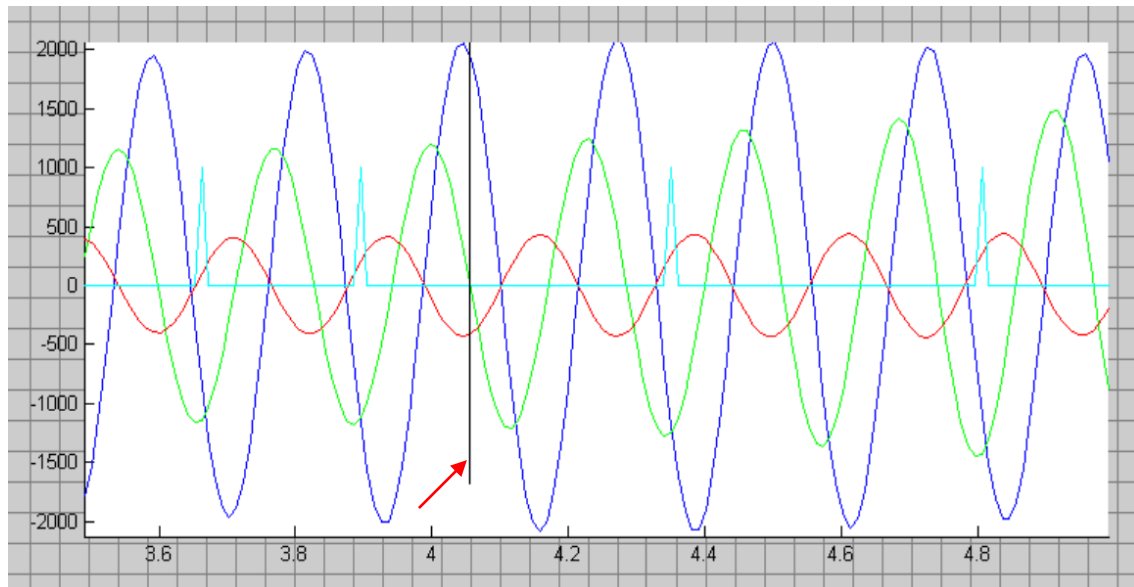


Gràfica 12: Acceleracions i pas d'aspa (X blau, Y verd, Z vermell, pas d'aspa Cian)



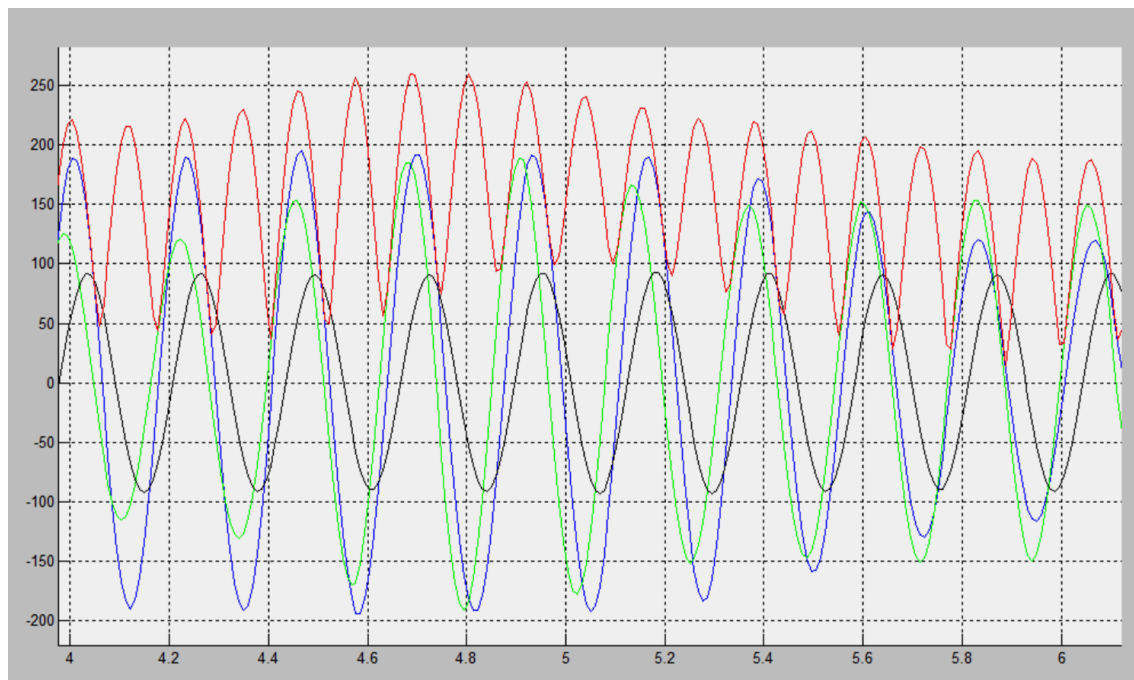
Gràfica 13: Acceleracions X en blau, Y en verd i la seva suma vectorial en vermell

Seguint amb el model teòric, per tenir un mòdul d'acceleració constant, els màxims d'un eix haurien de coincidir amb els zeros de l'altre. A més, el desfasament entre senyals hauria de ser de  $90^\circ$ . A la gràfica 14 comprovem que no es compleixen aquestes premisses.



**Gràfica 14: El pas per 0 de l'eix Y (verd) no coincideix amb el màxim de l'eix Y (blau)**

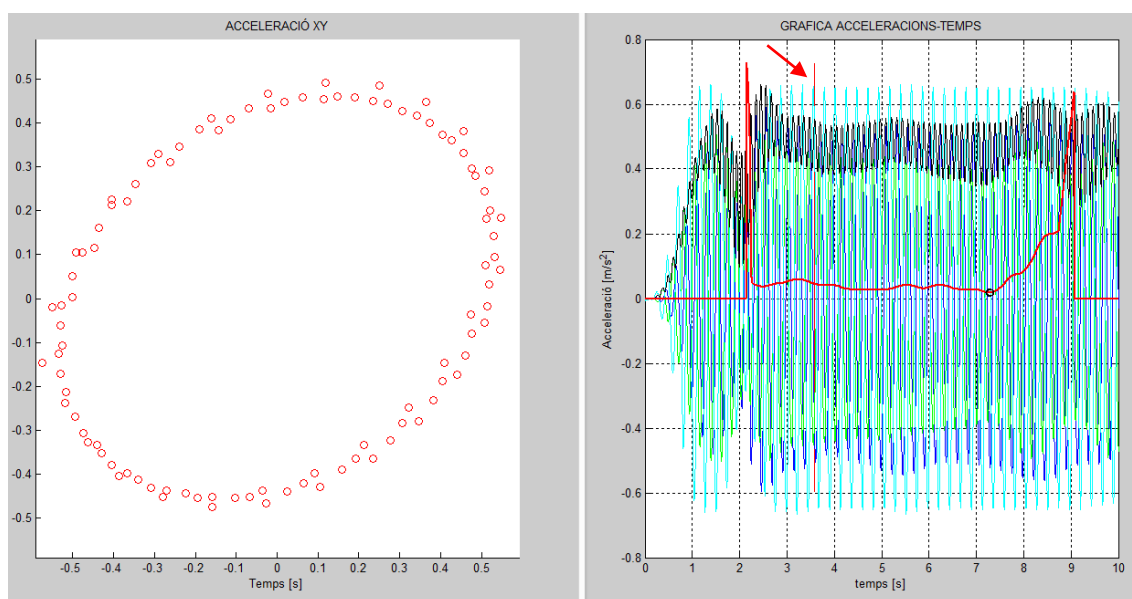
Malgrat aquestes discrepàncies amb el model teòric, hem fet un primer equilibrat a partir dels valors de les gràfiques anteriors. Quan tornem a mesurar les vibracions del ventilador, el resultat és que les acceleracions es redueixen prop del 80%, però les desviacions respecte al model teòric es veuen magnificades. A la gràfica 15 veiem que el desfasament entre els eixos X i Y s'ha reduït fins a quasi solapar-se les senyals.



**Gràfica 15: Pèrdua de desfasament X-Y al millorar l'equilibrat**  
(Accel. X: blau, accel. Y: verd, X+Y: vermell, pas d'aspa filtrat: negre)

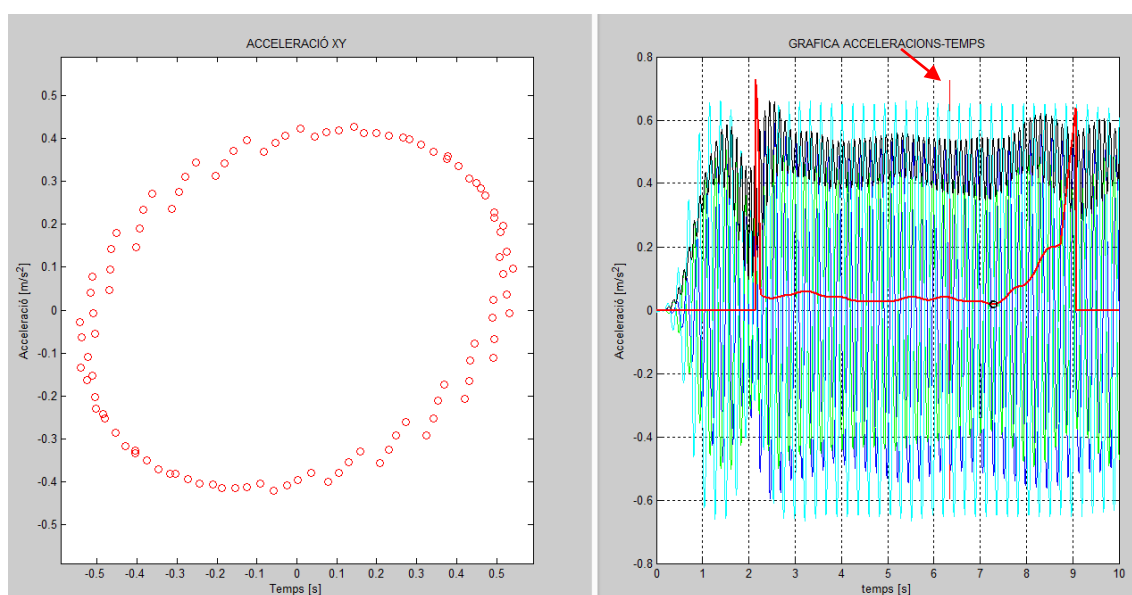
Hem intentat un segon equilibrat en base a aquets valors sense aconseguir reduir més les vibracions. La diferència entre el que està passant realment al ventilador i el model teòric no permet millorar l'equilibrat.

Per analitzar en profunditat la dinàmica del ventilador hem generat una funció amb Matlab que anomenem "evolució\_temps.m". A la part esquerra de la pantalla graficarà l'acceleració X contra la Y i a la part dreta ens indicarà amb una línia vertical vermella l'instant al que correspon. També hi podem veure el mòdul de l'acceleració resultant de sumar X i Y en color negre. De la figura 26 a la 28 podem observar seqüències de 4 voltes de les acceleracions del ventilador sense equilibrar. La corba vermella de la gràfica de la dreta és la sortida d'una funció (variància de fase) que més endavant detallarem. El petit cercle negre sobre aquesta corba indica el seu valor mínim.

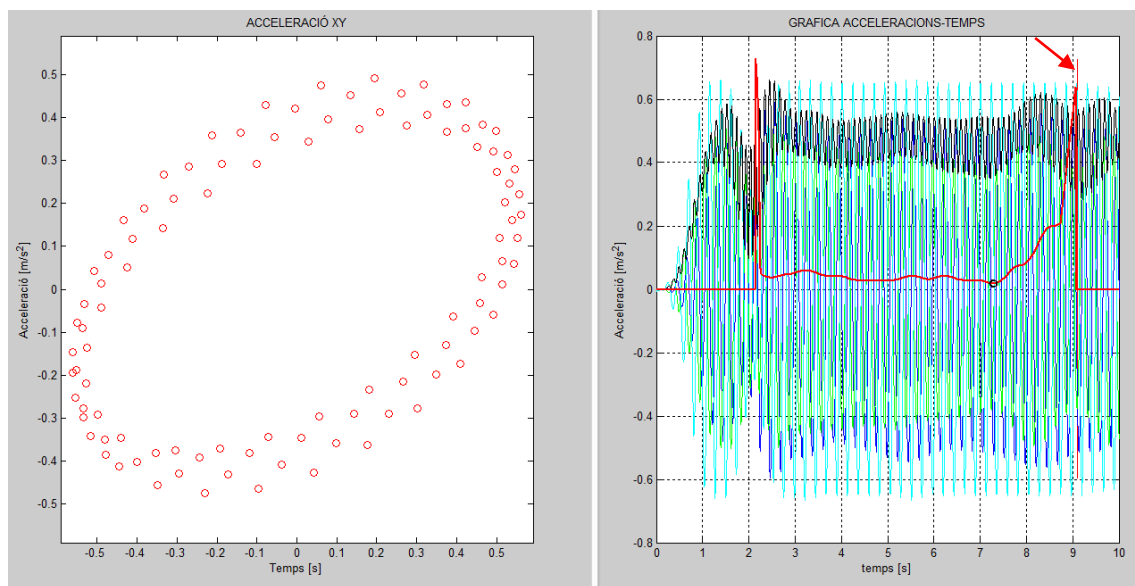


**Figura 26: Ventilador sense equilibrar (4 voltes anteriors al segon 3,6)**

Esquerra: accel. X contra Y. Dreta: La línia vermella vertical indica l'instant que s'està graficant  
En negre el mòdul de l'acceleració en el pla X-Y.



**Figura 27: Ventilador sense equilibrar (4 voltes anteriors al segon 6,3)**



**Figura 28: Ventilador sense equilibrar (4 voltes anteriors al segon 9,1)**

En les tres figures anteriors s'observa que les acceleracions no descriuen una trajectòria completament circular. Per tant els moviments del ventilador tampoc ho fan. Aquest fet l'hem pogut comprovar al graficar les velocitats i les posicions a partir de la integració de les acceleracions i les velocitats respectivament. Aquest efecte el podem atribuir a la diferent reacció dels ancoratges en una o altra direcció que hem descrit amb la fotografia de la Figura 24.

Entre les figures 30 i 33 repetim el procés amb el pes de prova al ventilador. Observem alteracions al mòdul de l'acceleració en el pla XY (en negre). Sobre el segon 4 perd intensitat i sobre el segon 9 n'augmenta. En tots dos casos s'incrementa la variabilitat del mòdul.

Per explicar aquest fenomen cal tenir en compte que al model teòric suposàvem que els moviments del ventilador responien únicament a la força exercida per la rotació de la massa en desequilibri. Però en realitat el ventilador s'inclina respecte a la vertical, fet que afegeix forces de component horitzontal com si d'un pèndul es tractés.

Per tenir una idea de la incidència de la força per l'efecte pèndul calia mesurar la inclinació del ventilador en moviment. Per fer-ho hem situat un punter làser al centre del ventilador que projecta sobre una diana amb escala mètrica al terra (figura 29). D'aquesta manera hem fet els mesuraments i càlculs de la Taula 3.

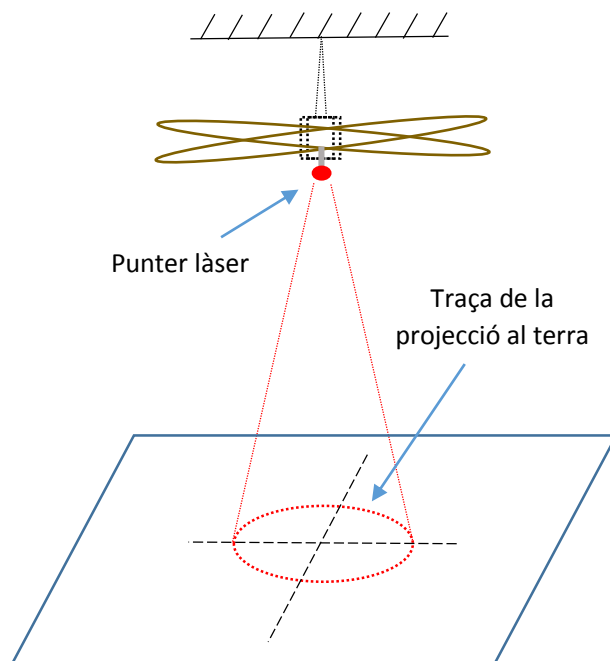


Figura 29: Muntatge experimental per mesurar la inclinació del ventilador en moviment

Taula 3: Mesuraments de l'angle d'inclinació del ventilador

Ventilador	Radi de màxima desviació $r$ [cm]	Angle $\alpha$ $\arctan(r/h)[^\circ]$	Acceleració horitzontal $g \cdot \sin(\alpha)$ [ $m/s^2$ ]	% sobre l'acceleració X-Y [ $m/s^2$ ]
<b>sense equilibrar</b> òrbita transitòria a la freq. de ressonància del conjunt	3 cm	$0,61^\circ$	$0,104 m/s^2$	
<b>sense equilibrar</b> òrbita estacionària a la freq. de rotació nominal	0,3 cm	$0,06^\circ$	$0,010 m/s^2$	$A(X-Y)=0,5$ $0,01/0,5=2\%$
<b>equilibrat</b> òrbita transitòria a la freq. de ressonància del conjunt	1,5 cm	$0,31^\circ$	$0,053 m/s^2$	
<b>equilibrat</b> òrbita estacionària a la freq. de rotació nominal	0,1 cm	$0,02^\circ$	$0,003 m/s^2$	$A(X-Y)=0,07$ $0,003/0,07=4\%$

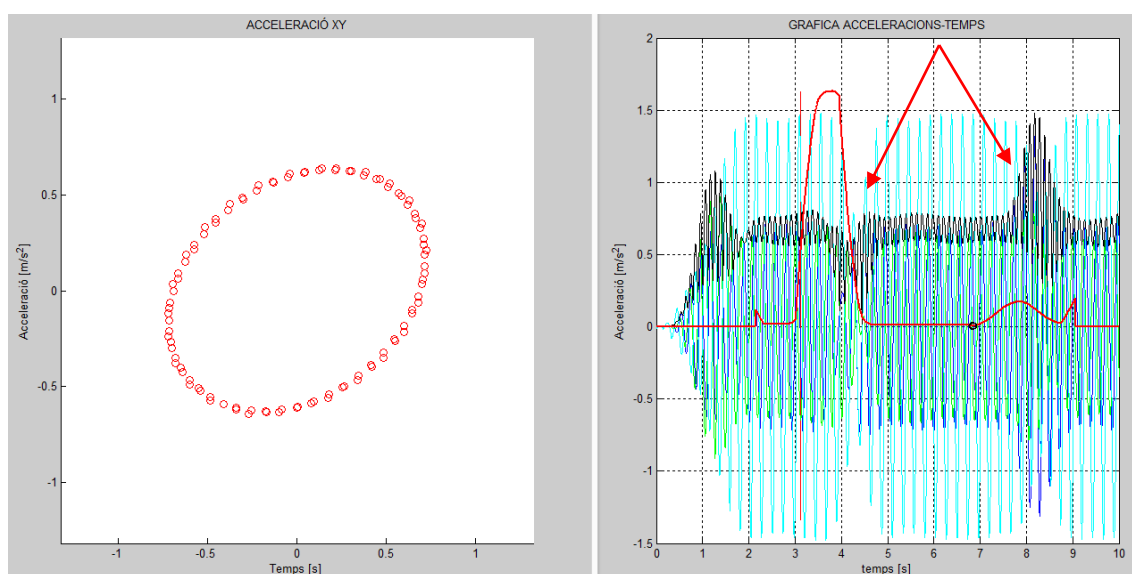
$h$ : Alçada de la subjecció del ventilador (distància al terra 2,80m)

$r$ : Distància de la projecció del punter làser al centre de la diana situada al terra.

$g$ : Acceleració gravetat ( $9,81m/s^2$ )

A priori sembla que l'efecte de la inclinació es podria negligir. Però quan el ventilador s'inclina, a més de les forces de gravetat que ens provoquen d'un 2 a un 4% de l'acceleració horitzontal, tindrem les forces de reacció de les subjeccions, que ja hem vist que no actuen en totes les direccions per igual i que no són lineals.

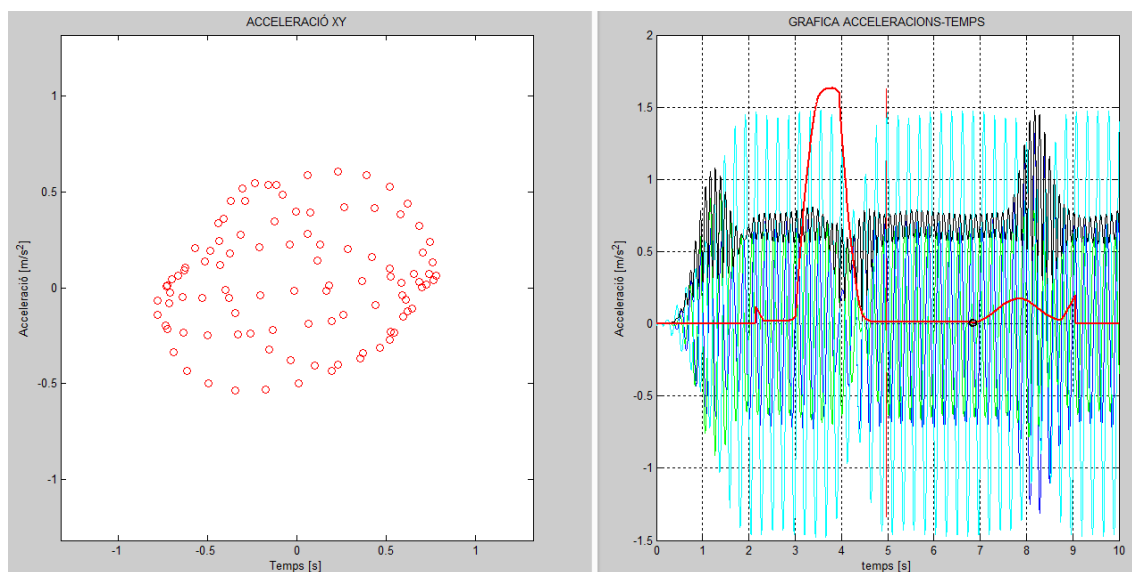
L'efecte pèndul sumat a als canvis en la força de les reaccions de les subjeccions ens introdueix un moviment força caòtic que ressonarà a la freqüència natural del conjunt i que és diferent a la freqüència de rotació. Així doncs, ambdues freqüències s'acoblen en un uns moments i es contraresten en altres, donant explicació a les variacions del mòdul de l'acceleració en el pla XY que assenyalarem amb fletxes vermelles a la dreta de la figura 30. A la part esquerra observem que les acceleracions en el pla XY durant els segons 2 i 3. En aquest cas descriuen una òrbita força circular i estable. En canvi, a la figura 31, corresponent als segons 4 a 5, s'observa una òrbita inestable.



**Figura 30: Ventilador amb pes de prova (4 voltes anteriors al segon 3,1)**

Esquerra: accel. X contra Y. Dreta: La línia vermella vertical indica l'instant que s'està graficant.

En negre el mòdul de l'acceleració en el pla XY.



**Figura 31: Ventilador amb pes de prova (4 voltes anteriors al segon 5)**



A la figura 32 tornem a observar una òrbita circular i estable, corresponent aproximadament als segons 6 a 7 de la senyal. I altre cop, en les 4 voltes anteriors al segon 8,7 (figura 33) tornem a observar una òrbita excèntrica i inestable.

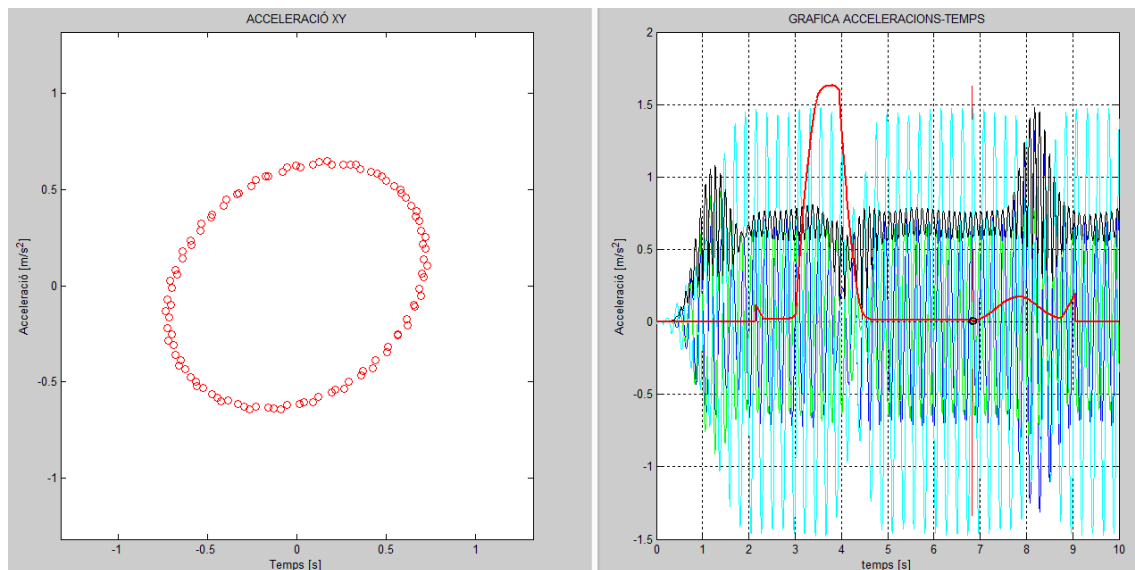


Figura 32: Ventilador amb pes de prova (4 voltes anteriors al segon 6,8)

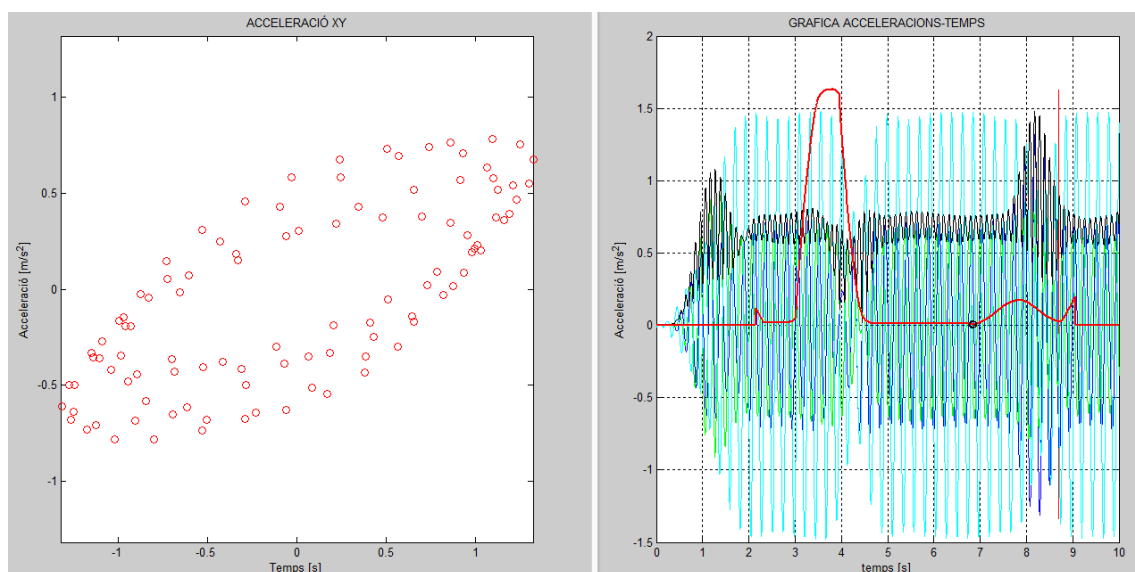
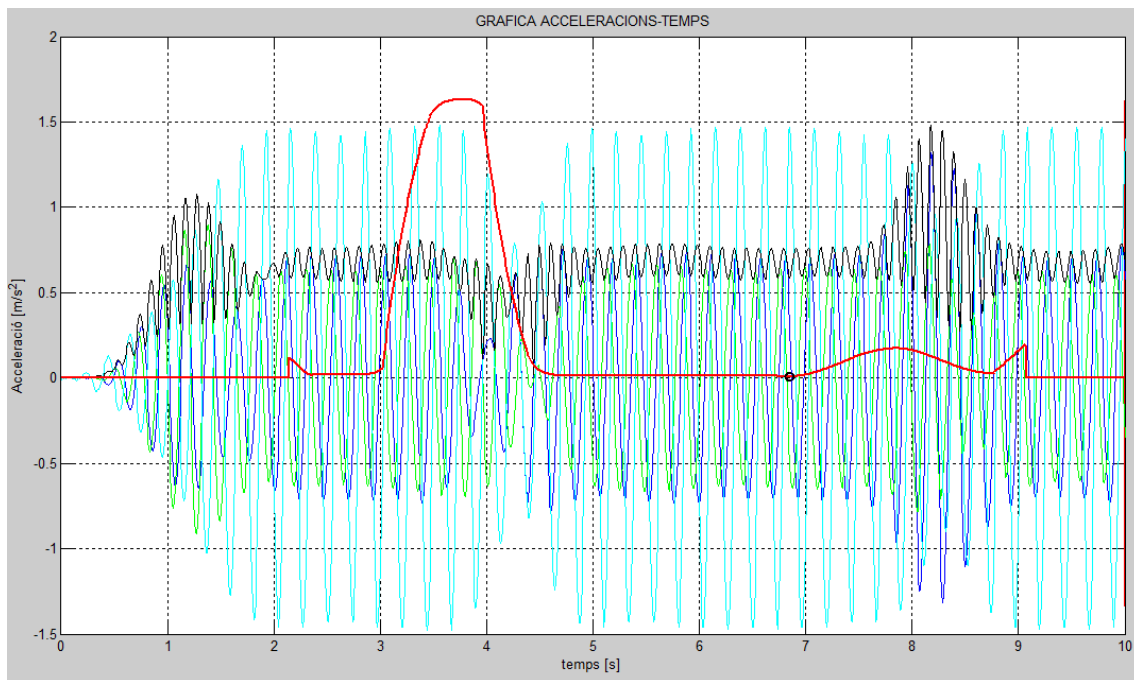


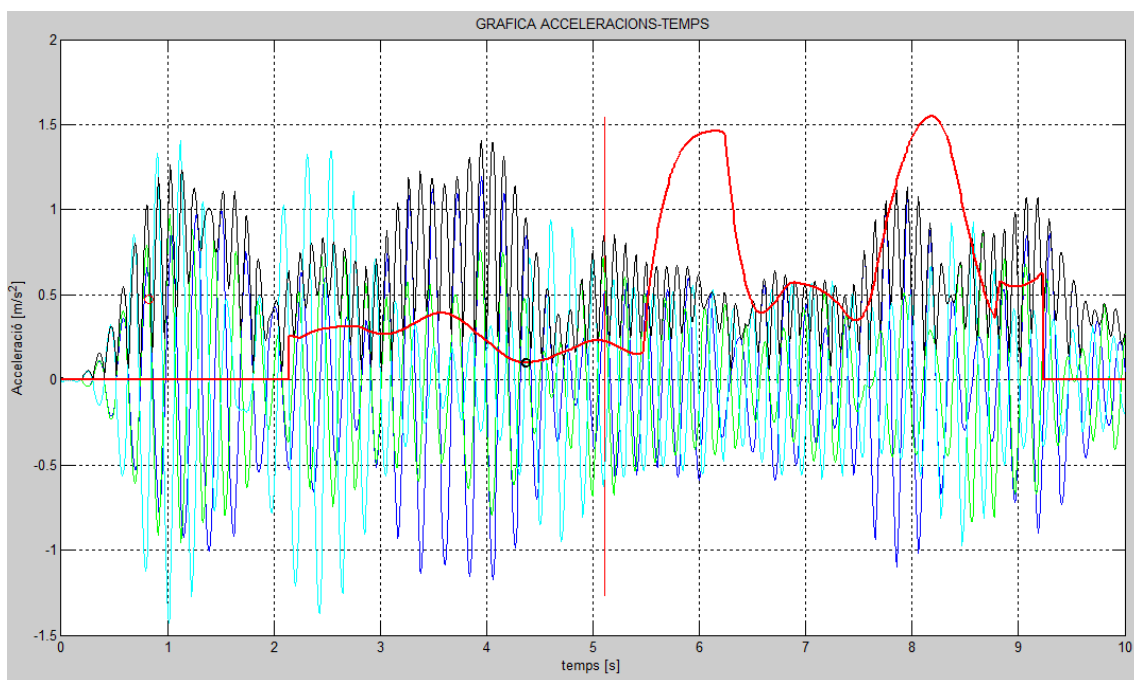
Figura 33: Ventilador amb pes de prova (4 voltes anteriors al segon 8,7)

A la gràfica 16 i 17 es comparen les dades de dos lectures consecutives amb el ventilador en les mateixes condicions (amb el pes de prova col·locat). Observem que en la segona lectura (gràfica 17) les acceleracions detectades són força irregulars, fet que ens generarà un moviment caòtic que no respon al model teòric, i per tant, no podrem obtenir valors per calcular les correccions del desequilibri a partir d'aquesta senyal.

No és habitual que ens apareguin resultats d'aquest tipus quan el ventilador està desequilibrat, és a dir, quan presenta acceleracions de l'ordre de  $0,5\text{m/s}^2$  o superiors. Si es dona el cas, caldrà repetir la lectura d'acceleracions per obtenir valors d'una estabilitat similar als de la gràfica 16.

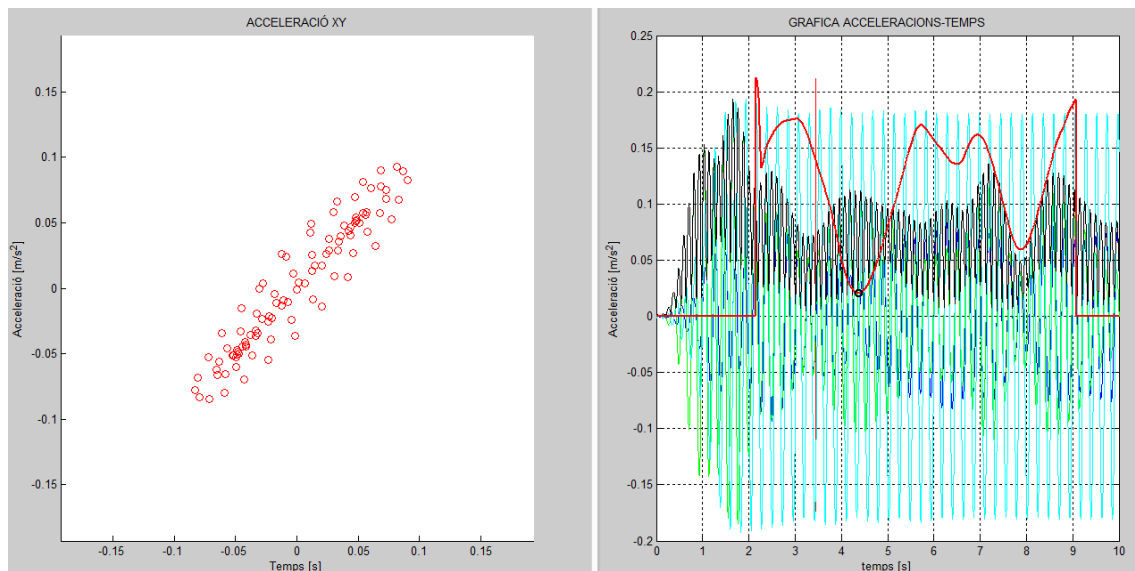


Gràfica 16: Ventilador amb pes de prova (mòdul de l'acceleració estable)



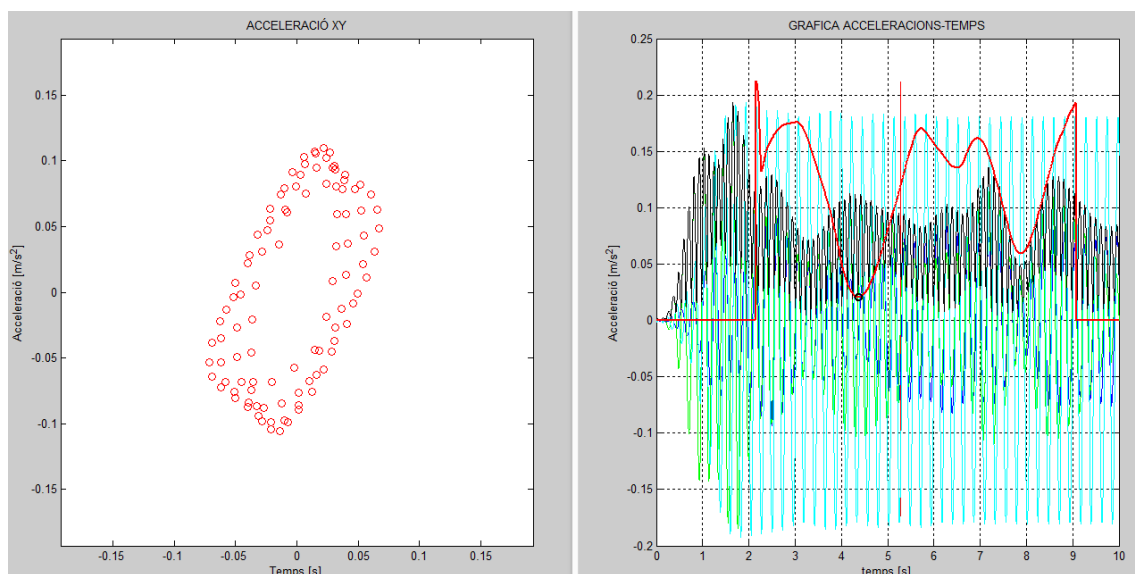
Gràfica 17: Ventilador amb pes de prova (mòdul de l'acceleració inestable)

Realitzem un primer equilibrat a partir de les dades més estables de la vibració original del ventilador (figura 26) i de les més estables amb el pes de prova (figura 32). El resultat és que la senyal presenta variacions al llarg del temps amb un comportament irregular com s'observa a les figures 34 a 36.

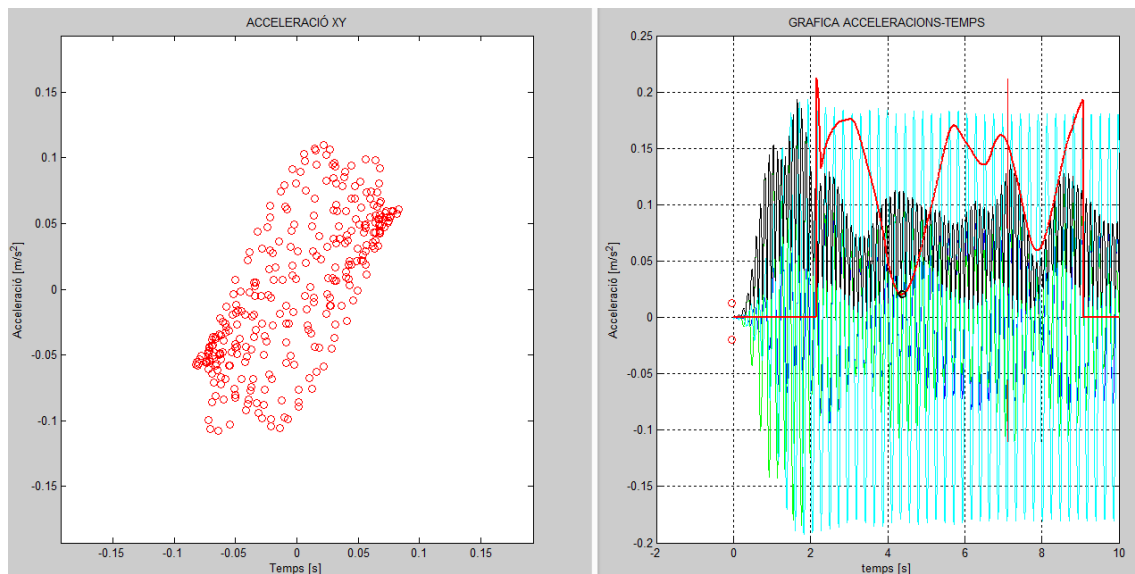


**Figura 34: Ventilador equilibrat (entre els segons 2,5 i 3,5)**

Esquerra: accel. X contra Y. Dreta: La línia vermella vertical indica l'instant que s'està graficant.  
En negre el mòdul de l'acceleració en el pla XY.



**Figura 35: Ventilador equilibrat (entre els segons 4,3 i 5,3)**



**Figura 36: Ventilador equilibrat (entre els segons 6 i 7)**

Cal destacar que amb la correcció dels contrapesos hem reduït les acceleracions inicials de 0,5 a 0,1  $\text{m/s}^2$ . És a dir, un 80%. Això comporta que el desequilibri de les aspes sigui menys dominant respecte a la resta de factors que ens aporten la vibració caòtica descrita anteriorment. Per tant, a mesura que corregim el desequilibri de les aspes, la senyal de les acceleracions es torna més irregular. Aquest serà el factor més important que limitarà la precisió de l'equilibrat.

Com a conclusió de l'anàlisi gràfic de les dades, hem vist que en les mesures de les acceleracions hi ha l'efecte del desequilibri de les aspes i per tant en podrem treure les dades que ens ajudin a corregir-lo. Però també hi ha forces que s'escapen al model teòric i que generen moviments i acoblaments de freqüències que emmascaren les dades que necessitem pel càlcul dels contrapesos. Així doncs, entre totes les dades d'acceleració que capturem, caldrà fer una selecció d'aquelles que realment responguin al desequilibri de les aspes.

## 7. Anàlisi de la senyal (elecció de valors òptims per al càlcul)

De tota la senyal capturada necessitem obtenir un únic valor d'amplitud i un altre de fase de l'acceleració en el pla XY. D'una banda sabem que per augmentar la precisió dels sensors podem fer la mitjana de lectures repetitives. D'altra banda sabem que la senyal té valors erràtics que no és corresponen al que pretenem mesurar i valors que si es corresponen al comportament del desequilibri d'un disc.

L'objectiu d'aquest apartat serà localitzar un tram de la senyal amb diverses voltes del ventilador que tinguin valors estables i responguin al model teòric. Posteriorment farem la mitjana dels valors d'aquestes voltes.

Per observació directa de les gràfiques anteriors podem intuir les zones més estables de la senyal per prendre les dades, però com pretenem que tots els càlculs els realitzi un microcontrolador, caldrà dissenyar un algoritme que mitjançant càlculs matemàtics localitzi les dades òptimes de la senyal.

Partim de la idea que les acceleracions produïdes pel desequilibri de les aspes responen al model teòric. Per tant, quan les dades de la senyal s'assemblin més a les previstes pel model teòric serà quan menys interferència per altres factors tinguem sobre les acceleracions.

Quines haurien de ser les característiques de la senyal segons el model teòric?

- 1- El mòdul de l'acceleració en el pla XY és constant.
- 2- El desfasament entre les senyals X i Y és de  $90^\circ$ .

Respecte a la primera característica, hem provat amb una funció que ens calcula la variància del mòdul al llarg del temps per un nombre de voltes  $n$   $\{Variància(t,n)\}$ . La idea és trobar la zona de la senyal on les properes "n" voltes tenen menor variància. Però en alguns casos, quan la freqüència natural del conjunt contraresta la vibració per la rotació, el mòdul de l'acceleració es fa petit reduint també la seva variància. Per aquest motiu hem de descartar aquesta opció.

Centrant-nos en la segona característica, comprovem que quan realment el desfasament s'apropa als  $90^\circ$ , les acceleracions descriuen una trajectòria força circular i el mòdul varia menys. Per tant, si trobem el desfasament de  $90^\circ$ , també ens apropem al model teòric pel que fa la menor variabilitat del mòdul. A més, quan les acceleracions es presenten més caòtiques, el desfasament de la senyal X respecte al Y s'avança o es retarda arribant a provocar canvis en el sentit de gir de l'oscil·lació del ventilador. Per tant, el desfasament entre les senyals X i Y serà un bon indicador per saber quan el ventilador vibra a causa del desequilibri.

Hem creat una funció amb Matlab "phase(senyal acceleració, retard)" que ens calcula com va incrementant l'angle a cada mostra de la senyal. La funció reparteix  $2\cdot\pi$  radians a les mostres situades entre 2 màxims consecutius de la senyal. D'aquesta manera la fase augmenta  $2\cdot\pi$  radians a cada volta. El paràmetre retard indica les mostres del principi de la senyal que s'han d'obviar perquè estan afectades pel retard del filtre. Amb els filtres d'ordre 197 i la Fs de 110Hz, el retard és d'uns 2 segons.

En el nostre cas, el sentit de gir va de l'eix Y al eix X. Per saber el desfasament haurem de restar la fase de X a la de Y. Com és possible que una de les senyals perdi alguna volta respecte a l'altra quan el ventilador té comportament erràtic, si les senyals s'allunyen més de mitja volta ( $\pi$  radians), corregirem l'angle una volta sencera ( $2\cdot\pi$  radians).

Per apropar-se al model teòric el desfasament ha de ser de  $\pi/2$  rad ( $90^\circ$ ). Calcularem la variància del desfasament respecte a  $90^\circ$  per cada valor de temps i per un nombre de voltes  $n$  (Variància\_fase(t,n)). Equació [3]

$$\text{Variància } \theta(m, n) = \frac{1}{n \cdot (T \cdot F_s)} \sum_{i=m}^{m+n(T \cdot F_s)} (\theta_i - 90^\circ)^2 \quad [3]$$

On:

$m$  és la variable número de mostra que conté al informació del temps ( $t = m/F_s$ )

$\theta_i$  el desfasament entre Y i X per cada mostra en graus.

$n$  la variable nombre de voltes

$T$  el període d'una volta

$F_s$  la freqüència de mostreig

$T \cdot F_s$  s'aproximarà a l'enter més proper i indicarà el nombre de mostres d'una volta

A la gràfica 18 es presenten els valors del desfasament Y-X (en graus). Veiem com l'algoritme ens detecta el conjunt de mostres que més s'apropa als  $90^\circ$ , indicant en vermell el tram que considera òptim. En cercle vermell sobre l'eix de les X ens indica la posició de la mostra on comença aquest tram. En color cian tracem la funció variància de desfasament respecte als  $90^\circ$ . Observem que el punt vermell és correspon al seu valor mínim. Aquest anàlisi es correspon a la senyal de la gràfica 19 amb el ventilador en el seu desequilibri original. Les acceleracions per l'efecte del desequilibri són dominants i la senyal força estable. És per això que la funció desfasament té valors baixos fins a partir del segon 8, en què també observem un increment de la variabilitat del mòdul.

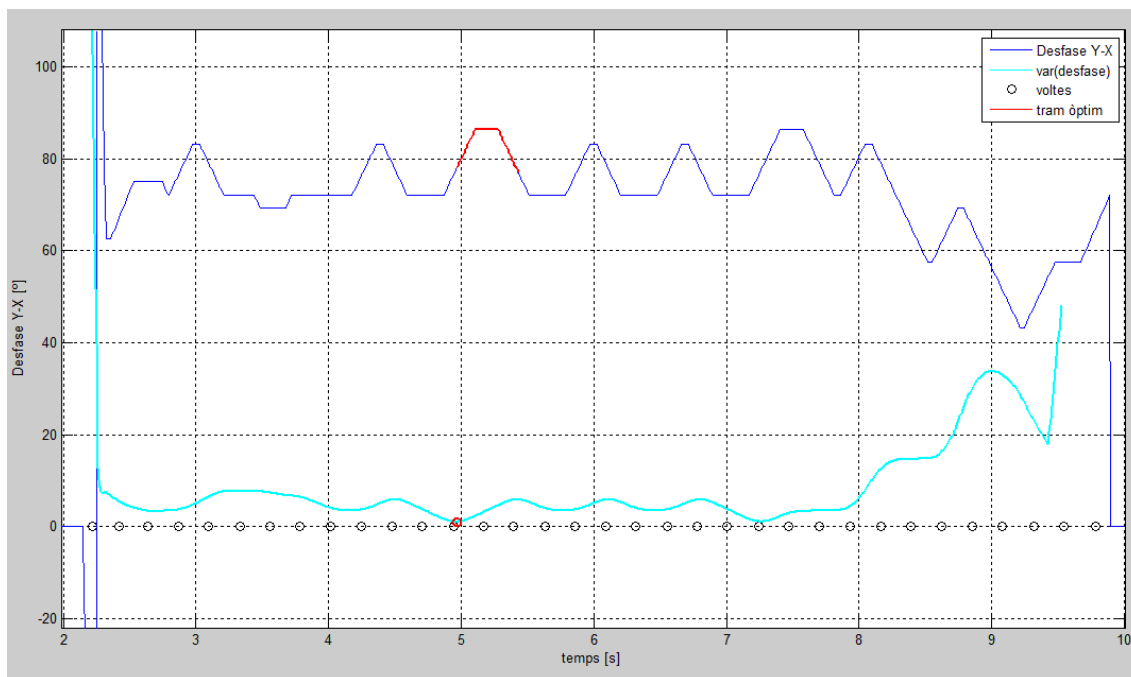
Calculem el mòdul de l'acceleració amb els valors òptims que hem determinat amb el ventilador desequilibrat: Mòdul acceleració =  $0,498\text{m/s}^2$

A la gràfica 20 i 21 repetim l'anàlisi anterior amb el ventilador equilibrat. Aquí s'observa un comportament més erràtic de la senyal. El desfasament Y-X només s'apropa als  $90^\circ$  a partir del segon 4,5, on la funció "variància desfasament" és fa mínima i l'algoritme ens marca en vermell el tram òptim d'aquesta senyal per a l'anàlisi. En aquest cas, havíem introduït el paràmetre " $n=4$ ", i per tant ens indica les 4 voltes consecutives amb desfasament més proper a  $90^\circ$ .

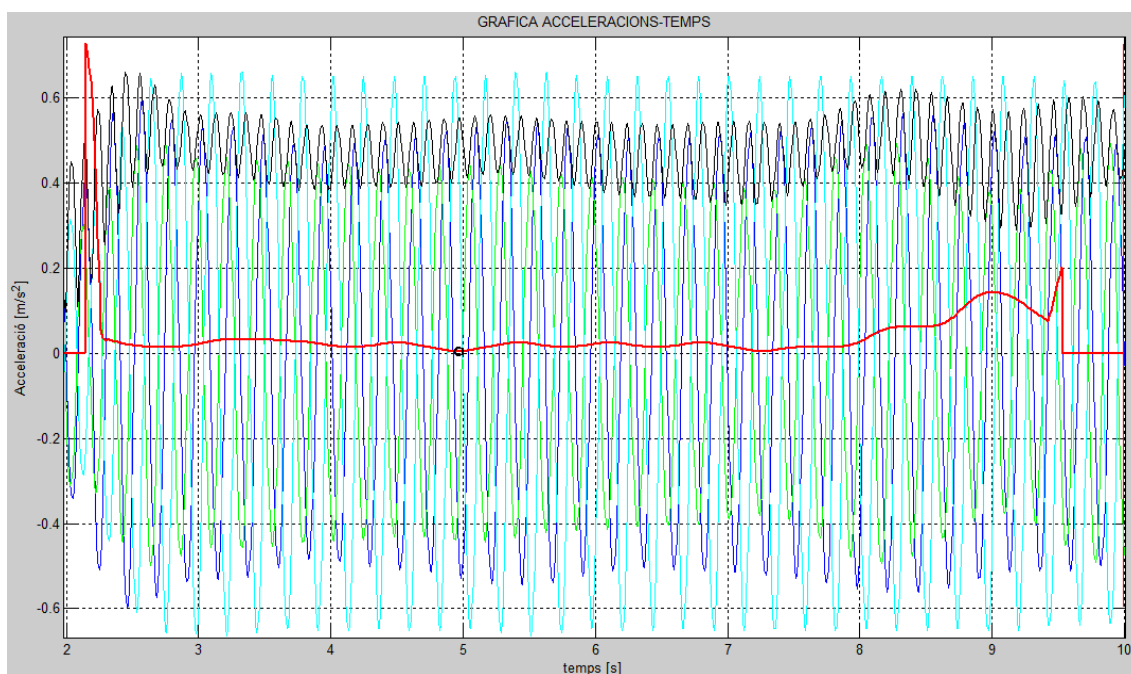
Calculem el mòdul de l'acceleració amb els valors òptims que hem determinat amb el ventilador equilibrat: Mòdul acceleració =  $0,077\text{m/s}^2$  (el 15% de les acceleracions originals)

Fem un segon equilibrat i repetim l'anàlisi amb les gràfiques 22 i 23. Hem reduït el mòdul de l'acceleració a  $0,035\text{m/s}^2$  (el 7% de les acceleracions originals)

A mesura que aconseguim reduir les vibracions pel desequilibri, la senyal es torna més inestable. El desfasament no aconsegueix superar els  $50^\circ$ . Fins i tot l'eix X perd una volta respecte al Y entre els segons 5 i 7 (gràfica 24). A la figura 37 es mostra la trajectòria de les acceleracions en aquest temps.

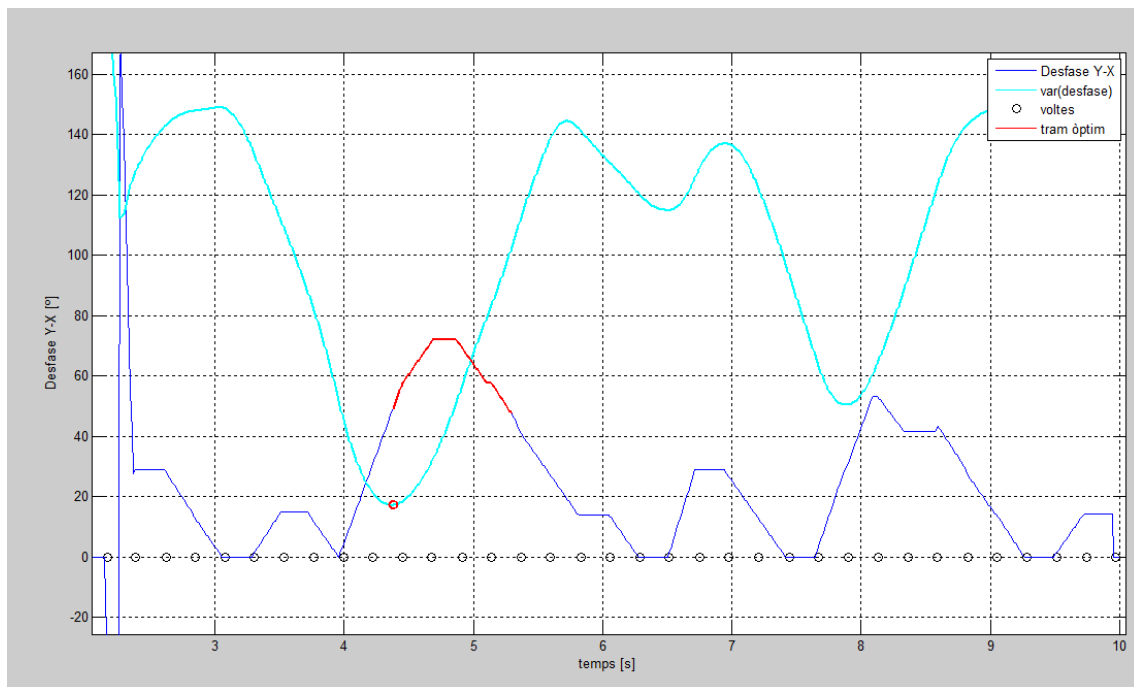


**Gràfica 18: Anàlisi de la senyal de la gràfica 19**  
Desfasament Y-X en blau. Funció "Variància\_fase(t,n)" en cian

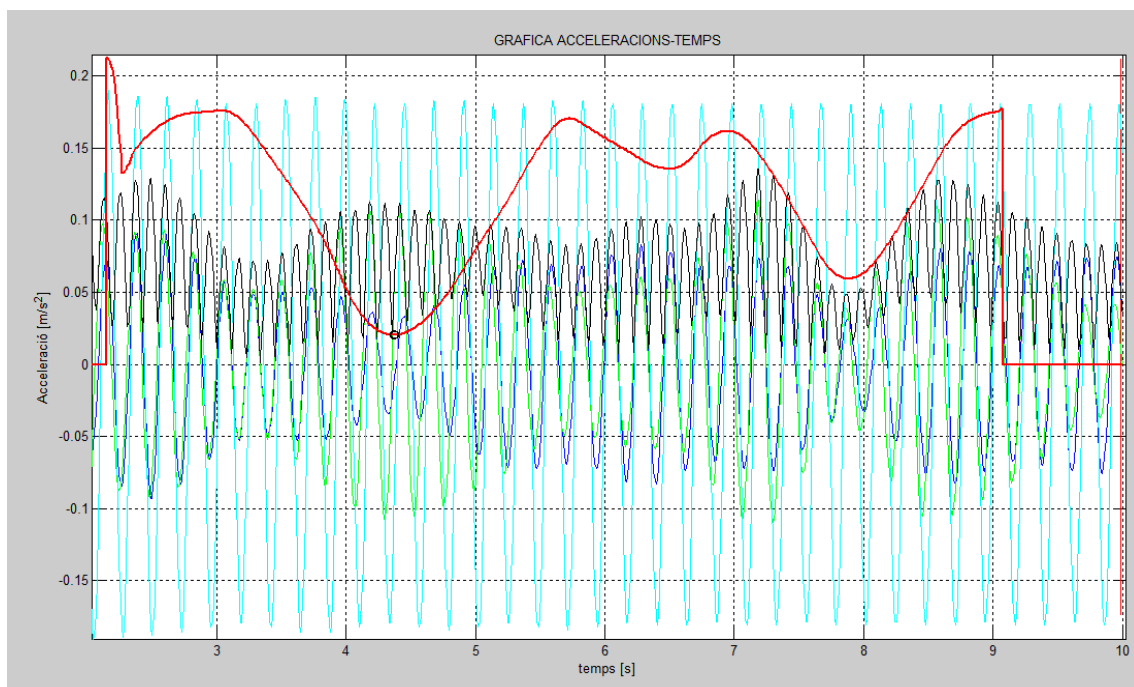


**Gràfica 19: Ventilador en estat original. Funció "Variància\_fase(t,n)" en vermell**

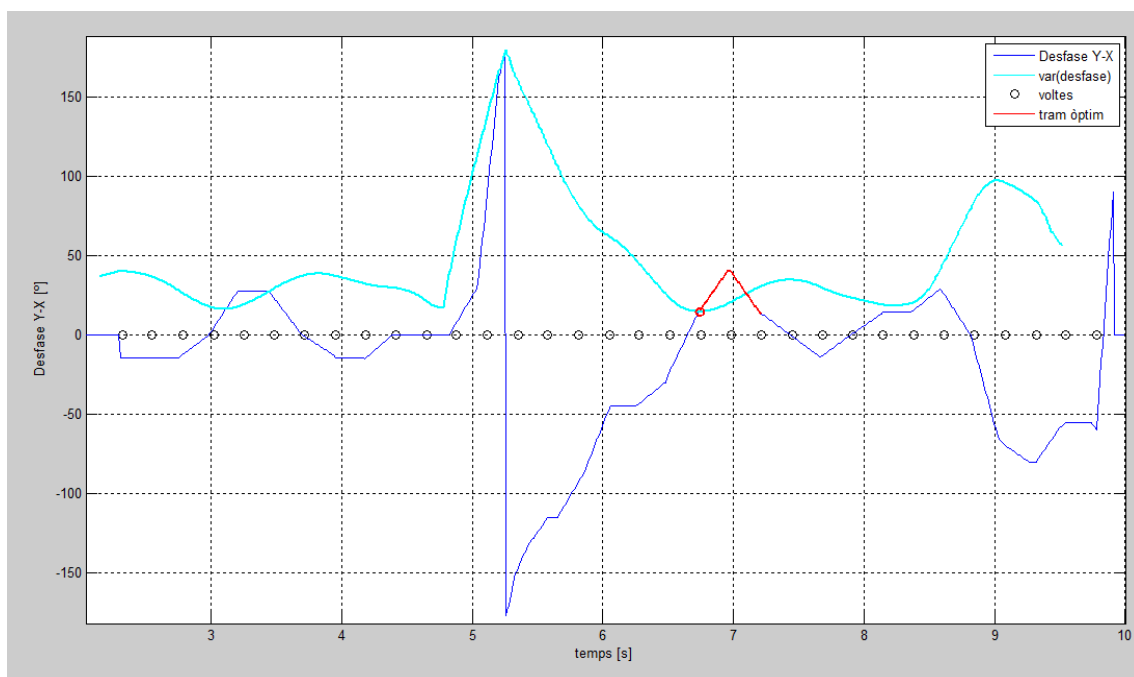




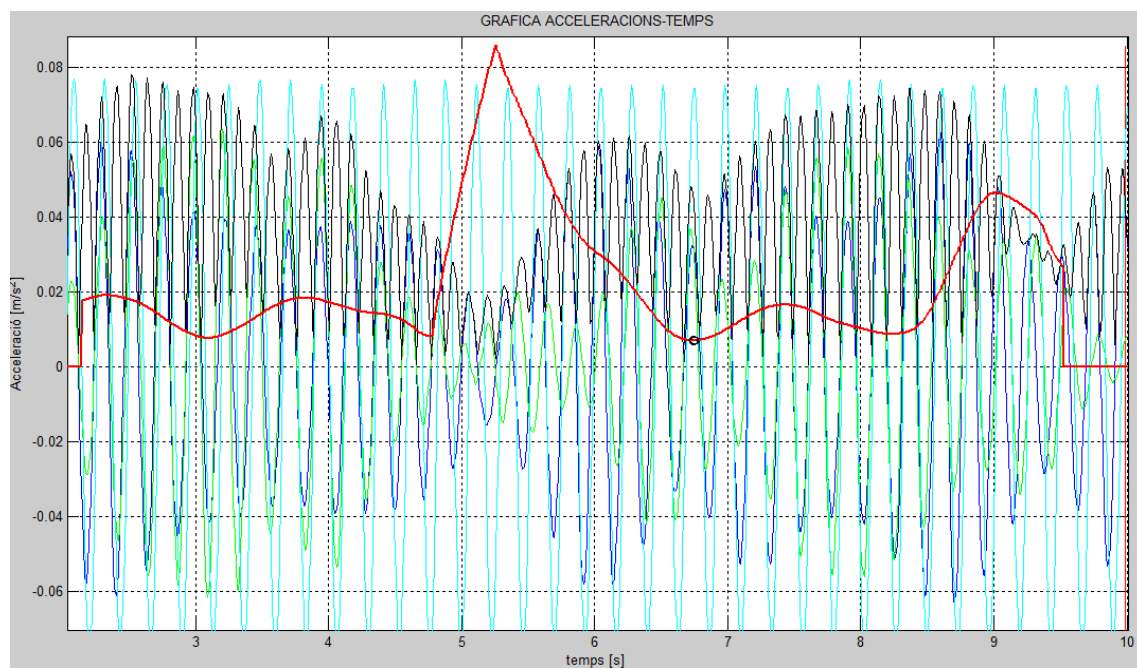
**Gràfica 20: Anàlisi de la senyal de la gràfica 21**  
Desfasament Y-X en blau. Funció "Variància\_fase(t,n)" en cian



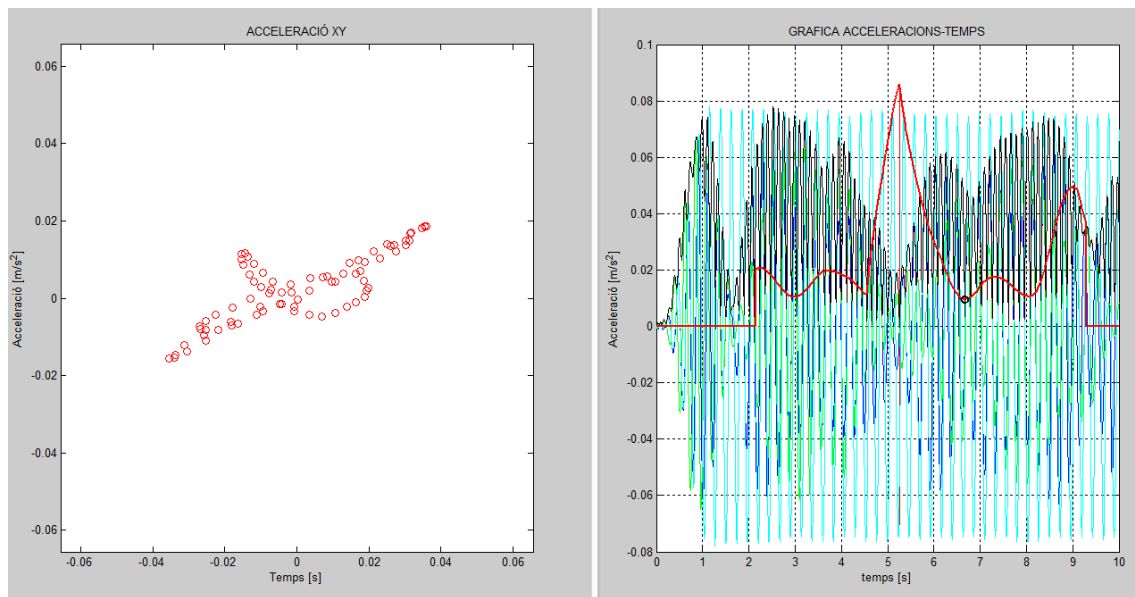
**Gràfica 21: Ventilador equilibrat. Funció "Variància\_fase(t,n)" en vermell**



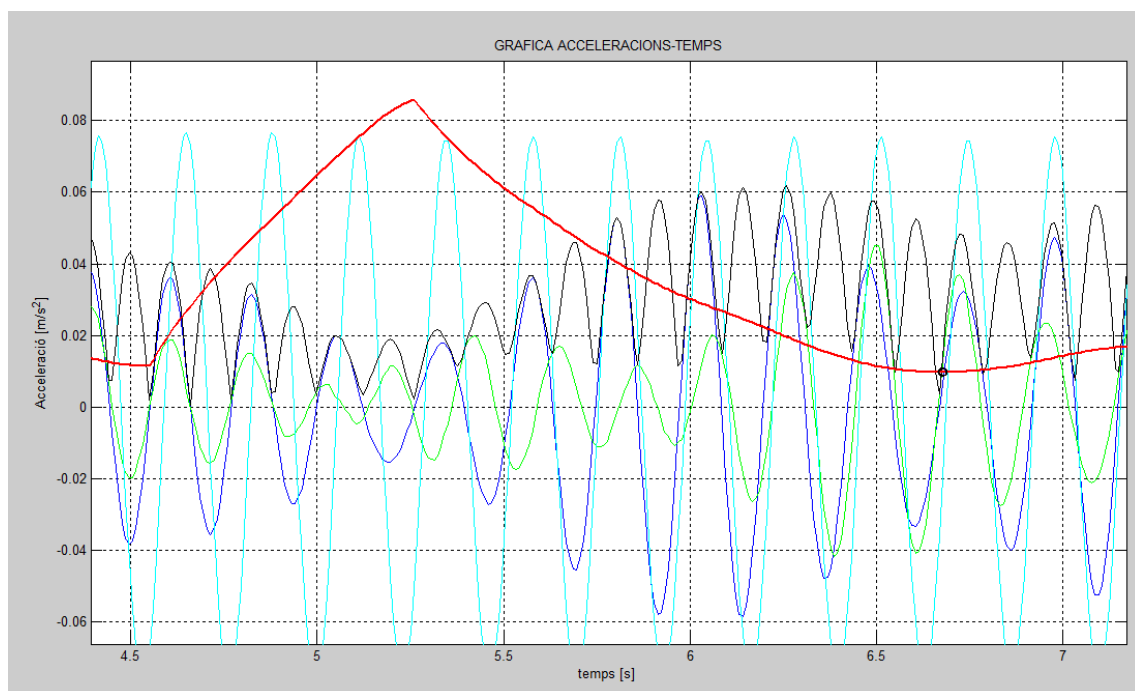
**Gràfica 22: Anàlisi de la senyal de la gràfica 23**  
Desfasament Y-X en blau. Funció "Variància\_fase(t,n)" en cian



**Gràfica 23: Ventilador reequilibrat. Funció "Variància\_fase(t,n)" en vermell**



**Figura 37: Ventilador reequilibrat, acceleracions petites ( $0,04\text{m/s}^2$ ), de direcció irregular**



**Gràfica 24: Ventilador reequilibrat. L'eix X per un període respecte al Y**

Com a conclusió d'aquest apartat podem dir que amb l'algoritme que localitza dins la senyal els valors que es comporten com el model teòric millorem la precisió dels càlculs. Tot i així, per factors de construcció del ventilador, serà difícil reduir les vibracions per sota del 10%.

## 8. Càlcul del període, l'amplitud i la fase de l'acceleració

### a) Període (T):

Per a calcular el període (T) es basem en la senyal del pas d'aspa perquè és la més homogènia. Comptem el nombre de polsos i el dividim pel temps de la senyal (10 segons). La freqüència de la senyal serà  $1/T$ . A partir de la freqüència triarem el filtre que millor s'hi adapti. Si fos massa elevada pels filtres dissenyats enviàriem un missatge a l'usuari indicant error.

Per evitar cometre l'error de mesurar el període mentre el ventilador encara està adquirint velocitat, hem fet una funció per calcular la variància del període al microcontrolador que no ens deixarà adquirir senyal fins que el període sigui estable.

### b) Amplitud (Mòdul):

En totes les lectures d'acceleracions acabarem obtenint el valor eficaç de l'amplitud (mitjana quadràtica o RMS -Equació [4]).

$$x_{\text{RMS}} = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} = \sqrt{\frac{x_1^2 + x_2^2 + \dots + x_N^2}{N}} \quad [4]$$

Partim del mòdul de l'acceleració en el pla XY, on tots els valors ja són positius, i farem la mitjana de les mostres òptimes per al càlcul. Preferim prendre el valor RMS enlloc de l'acceleració en l'instant en que detectem el pas d'aspa perquè ens farà la mitjana de totes les mostres durant "n" voltes, assolint una major precisió i responent millor al model teòric, segons el qual aquest mòdul hauria de ser constant en totes les mostres.

Pel que fa al número de voltes per a calcular les mitjanes que definim amb el paràmetre "n" recomanem situar-lo entre 2 i 5. El motiu és que quan la senyal és irregular ens costarà trobar més de 3 voltes consecutives que responguin al model teòric i no incorporin error als càlculs.

Per si tenim errors de linealitat, afegirem al càlcul una constant K que ens magnificarà o reduirà la força que hauran d'exercir els contrapesos. En principi li assignarem el valor 1. Més endavant, quan s'hagin fet proves amb diferents ventiladors, es pot mirar de trobar una relació que permeti escalar-la automàticament. Les variables a tenir en compte podrien ser el percentatge de variació de l'acceleració que ens provoca el pes de prova respecte a la situació inicial del ventilador i la excentricitat de l'òrbita descrita per les acceleracions.

## c) Fase (Angle de l'acceleració quan detectem el pas d'aspa):

Al posar els sensors al ventilador hem alineat l'eix X positiu de l'acceleròmetre amb el sensor òptic. D'aquesta manera, quan tenim un màxim en la senyal de pas d'aspa sabem que les aspes es troben alineades amb els eixos X i Y.

Es tracta de mesurar l'angle de l'acceleració en aquest instant (indicat amb cercles vermells a la gràfica 25). L'angle l'obtenim a partir de l'arctangent de l'acceleració en Y i en X. Tant Matlab com les llibreries del microcontrolador incorporen la funció  $\text{atan2}(Y,X)$  que ens retorna l'angle entre  $-\pi$  i  $+\pi$  radians.

L'angle resultant d'aquesta funció és el de l'acceleració real del ventilador en l'instant del pas d'aspa, però la direcció d'aquesta acceleració ve condicionada per les diferents forces de reacció dels ancoratges en una o altra direcció. En realitat, la massa en desequilibri està "estirant" en una altra direcció, però l'acceleració es veu limitada en la direcció on els ancoratges tenen més reacció, i per tant l'angle de l'acceleració real no assenyalava exactament la direcció del desequilibri de masses.

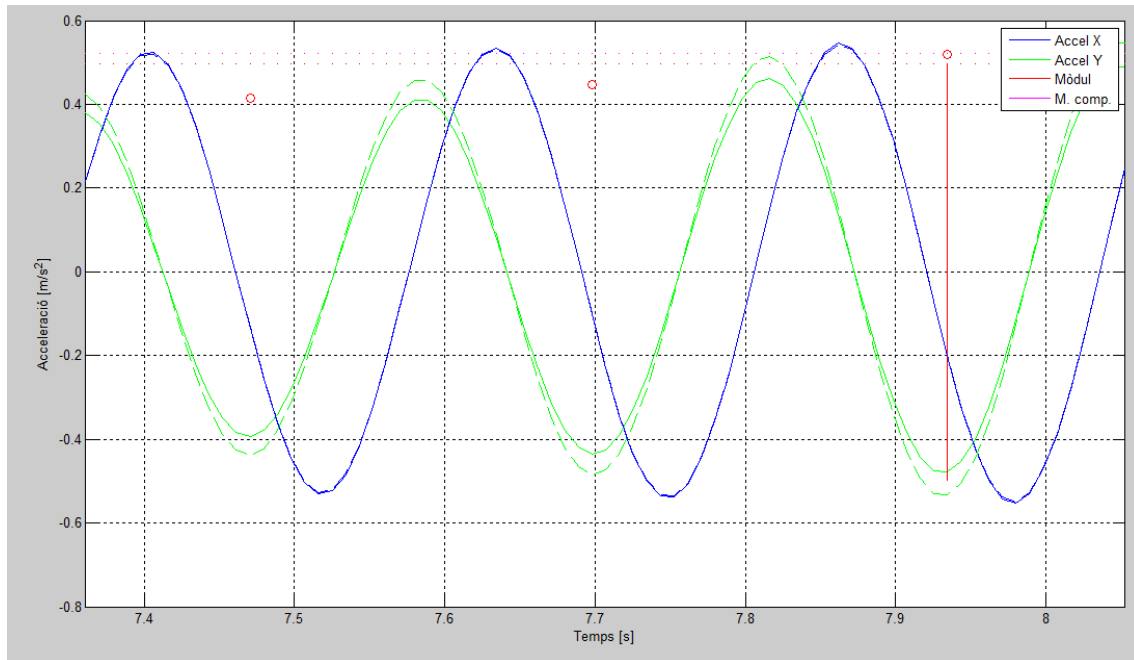
Aquesta diferència entre l'acceleració del ventilador i la que realment provoca el desequilibri de masses no és tant rellevant quan calculem el mòdul. En aquest cas captem informació de tota la volta i fem la mitjana de totes les mostres. En canvi, en el cas de l'angle ens podria introduir errors considerables i per això caldrà fer una correcció. D'una banda perquè hem de llegir l'angle en un moment concret, i per tant només disposarem d'un valor per volta alhora de fer la mitjana. D'altra banda, la lectura de l'angle es fa sempre quan l'aspa amb el reflector passa sobre el sensor òptic. Per tant, el moment en què fem la lectura dependrà d'un factor tant aleatori com és la direcció en què situarem els sensors.

Corregirem l'angle mesurat de l'acceleració fent un nou apropament al model teòric, el qual únicament respon als desequilibris del disc de rotació. Segons aquest model, l'amplitud d'acceleració en l'eix X i Y hauria de ser la mateixa. Per tant, abans de calcular l'angle amb la funció  $\text{atan2}(Y,X)$  escalarem les senyals per igualar-ne l'amplitud com si les reaccions de subjecció en els dos eixos fossin iguals.

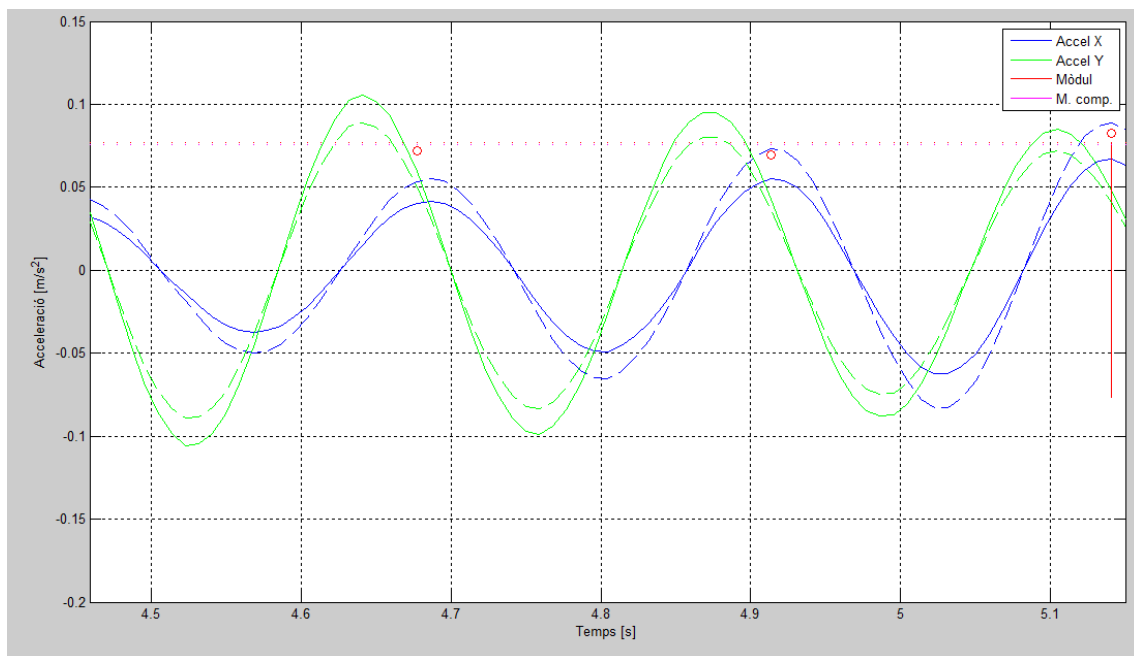
Per a escalar les senyals farem la mitjana dels dos valors màxims de l'acceleració en X i en Y anterior i posterior al pas d'aspa. Dels dos valors obtinguts anteriorment, prendrem el més gran com a factor d'escala. Després escalarem sobre aquest màxim les dos mostres (X i Y) que usem per calcular l'angle. La senyal de més amplitud es quedarà igual, mentre que la que es veia més limitada per les reaccions dels ancoratges, quedarà amplificada per igualar a la de major amplitud.

La gràfica 25 ens presenta les mostres òptimes de tres voltes ( $n=3$ ) sobre les que calcular el mòdul i la fase de l'acceleració del ventilador en el seu estat inicial. En línia contínua es tracen les acceleracions amb la seva amplitud original. En línia discontinua del mateix color s'han escalat les senyals a la mitjana màxima de les 3 voltes. En aquest cas veiem poca variació perquè al estar el ventilador desequilibrat, les forces dominants són les de la rotació del desequilibri de masses. Per tant, respon millor al model teòric i les amplituds de X i de Y són similars. L'angle mesurat a partir de les senyals originals és de  $-108,29^\circ$ , i l'angle mesurat amb les senyals escalades és de  $-105,76^\circ$ .

En canvi, a la gràfica 26, amb el ventilador equilibrat i les acceleracions reduïdes a un 15%, l'amplitud de X és prop de la meitat de la de Y. L'angle mesurat a partir de les senyals originals és de  $43,25^\circ$ , i l'angle corregit amb les senyals escalades és de  $35,48^\circ$ . És una diferència important perquè afectarà a la posició dels contrapesos.



Gràfica 25: Mostres òptimes i correcció amplitud. Ventilador sense equilibrar



Gràfica 26: Mostres òptimes i correcció amplitud. Ventilador equilibrat

Per resoldre tots els càlculs d'aquest apartat hem creat la funció "analisi\_senyal.m" a la qual li indiquem per paràmetre el nom de l'arxiu a analitzar, les mostres que conté, la Fs, si cal filtrar la senyal i el nombre de voltes sobre les que farem el càlcul. La funció ens retorna el període de la senyal, el mòdul de l'acceleració i l'angle amb la correcció descrita anteriorment.

## 9. Càlcul dels contrapesos per a l'equilibrat (massa i posició)

Hem creat la funció "calcul\_contrapesos.m" amb Matlab que fa els càlculs d'aquest apartat i ens genera la sortida per pantalla de la Figura 38.

### a) Comprovació de l'alineació del sensor òptic:

Per a saber com reacciona el ventilador al desequilibri fem una segona lectura amb un pes de prova a l'aspa que té el reflectant. L'acceleració es mesura en el moment en què aquesta aspa passa del sensor òptic, és a dir, quan està alineada amb l'eix X positiu. Per aquest motiu, la col·locació del pes de prova no hauria de modificar la component Y de l'acceleració original. Si detectem que varia per sobre d'una tolerància, ens estarà indicant que el sensor òptic no està correctament alineat amb l'eix X de l'acceleròmetre. Si això passa, el programa ens avisarà indicant la desviació en graus per tal que puguem corregir la posició del sensor i repetir les mesures. El càlcul es realitza amb l'equació [6].

$$\text{Desviació sensor òptic} = \text{asin}\left(\frac{\Delta A_{ppY}}{\Delta A_{pp}}\right) \quad [6]$$

On:

$\Delta A_{pp}$  és la variació entre el mòdul de l'acceleració original i el del pes de prova

$\Delta A_{ppY}$  és la seva component en l'eix y

### b) "massa equivalent"

La variació en l'acceleració la provoca la força del pes de prova, la qual calculem amb l'equació [7]. A partir de la força exercida pel pes de prova i l'acceleració que provoca, podem calcular la "massa equivalent del ventilador" amb l'equació [8].

$$F_{pp} = \text{massa pes prova} \cdot \text{Radi aspa} \cdot \omega^2 \quad [7]$$

$$M_{eq} = \frac{F_{pp}}{\Delta A_{ppx}} \quad [8]$$

On:

$F_{pp}$  és la força que genera el pes de prova

$(\text{Radi aspa} \cdot \omega^2)$  és l'acceleració normal que té el pes de prova en rotació

$M_{eq}$  és la "massa equivalent" del ventilador

La massa equivalent ens dona un valor d'uns 3,6kg. En aquest cas hem pesat el ventilador i la seva massa real és de 4,5kg. És a dir, la resistència a canviar la velocitat és inferior a la que correspondria per la seva massa real. Això pot ser degut a què en el moviment oscil·latori és sumen les forces del pes de prova amb les de l'efecte pèndul i les de les reaccions de les subjeccions, doncs la posició del ventilador té un desfasament de 180° respecte a la direcció de l'acceleració.



## c) Mòdul de la força que han de fer els contrapesos

La força que han de fer els contrapesos ha de tenir el mateix mòdul que la força que provoca les acceleracions en l'estat original del ventilador. Podem calcular-la amb l'equació [9].

$$F_c = M_{eq} \cdot \text{Acceleració inicial} \quad [9]$$

On:

$F_c$  és la força que generaran els contrapesos

Acceleració inicial és la que hem mesurat abans d'equilibrar

## d) Massa dels contrapesos

D'acord amb l'equació [10], la força que exerciran els contrapesos ( $F_c$ ) depèn de 3 variables: la massa del contrapès ( $m_c$ ), la distància on es col·loqui ( $R$ ) i la velocitat angular ( $\omega$ ):

$$F_c = m_c \cdot An(m_c) = m_c \cdot R \cdot \omega^2 \quad [10]$$

La  $\omega$  ve determinada pel ventilador. El valor de  $R$  màxim serà la longitud de l'aspa. La  $m_c$  la podem triar entre els pesos de les monedes de la figura 15. Per a cada eix el programa comprovarà si col·locant el contrapès més petit a l'extrem de l'aspa és suficient per generar la força que volem. En cas negatiu, agafarem el següent pes. I així successivament fins a trobar el pes suficient per a cada eix. Si no disposéssim de prou pesos donaríem un avis a l'usuari.

## e) Posició dels contrapesos

Partim de l'equació [10] i descomponem la força dels contrapesos en els dos eixos d'acord amb les equacions [11]

$$F_c x = F_c \cdot \cos(\theta(A_i)); F_c y = F_c \cdot \sin(\theta(A_i)) \quad [11]$$

on:

$\theta(A_i)$  és l'angle de l'acceleració en l'estat original del ventilador

Si aïllem  $R$  de l'equació [10] tenim:

$$R = \frac{F_c}{m_c \cdot \omega^2}$$

Per tant, la distància dels contrapesos al centre de gir l'obtindrem de les equacions [12]

$$R_x = \frac{F_c x}{m_{cx} \cdot \omega^2}; R_y = \frac{F_c y}{m_{cy} \cdot \omega^2} \quad [12]$$

## 10. Millora de la precisió i vibracions residuals

S'ha realitzat un equilibrat del ventilador amb la metodologia descrita en aquest apartat obtenint una reducció de les acceleracions de 90% (hem passat de 0,47 a 0,046 m/s<sup>2</sup>). Per intentar millorar la precisió hem fet petits desplaçaments en els contrapesos per intentar corregir les acceleracions residuals, les quals hem reduït fins a 0,03m/s<sup>2</sup>. Per sota d'aquest valor la senyal d'acceleració és molt irregular (Figura 39) i no s'aconsegueixen lectures repetitives. Per tant resulta impossible saber cap a on hem de desplaçar els contrapesos per reduir aquesta vibració que considerarem "residual".

Sabent la posició on els contrapesos ens aconsegueixen el millor equilibrat, hem modificat el valor de K (constant que havíem introduït per corregir errors de linealitat) a 1,08 de manera que la funció "càlcul\_contrapesos.m" ens dona ara la posició dels contrapesos on aconseguim el millor equilibrat. A la figura 38 presentem la sortida de pantalla de la funció que ens indica la massa i posició dels contrapesos amb els que obtenim l'equilibrat òptim.

```
>> calcul_contrapesos('./senyal/vibracio_original.txt', './senyal/vibracio_pes_prova.txt', 1100, 110, 1, 3)

-----
Analisi senyal fitxer: ./senyal/vibracio_original.txt
Periode           = 0.234375 s
Mòdul             = 0.470805m/s^2    angle           = -64.86°
Mòdul corregit    = 0.492671m/s^2    angle corregit  = -61.73°
-----

Analisi senyal fitxer: ./senyal/vibracio_pes_prova.txt
Periode           = 0.240057 s
Mòdul             = 0.730168m/s^2    angle           = -24.46°
Mòdul corregit    = 0.735264m/s^2    angle corregit  = -23.92°
-----

----- Càlcul contrapesos -----
Mòdul inicial: 0.470805 m/s^2   Angle inicial: -61.7°   Periode inicial: 0.2344 s
Mòdul pes prova: 0.730168 m/s^2   Angle contrap: -23.9°   Periode contrap: 0.2401 s
Comprovació variació accel. Y per contrapes: 0.118657   Variació accel. X: 0.444512
Si variació accel. Y >> 0 ==> avançar sensor òptic. En cas contrari retrassar.
desviacio_sensor_optic: 14.95°   Graus per mostra: 13.96°

Força pes de prova [Fpp = mpp·(R·w^2)]: 1.342727 N   Massa equivalent: 2.92 kg
Força per contrarestar: Fc = Ao·(Fpp/varAx pel pes prova): 1.374036 N

VARIACIÓ DEL MÒDUL CORRECTA: 55.09% (entre 30% i 100%)
Factor correcció: k = 1.08

* * * SORTIDA * * *
Contrapès X: 2.30 g      Radi X: -36.45 cm
Contrapès Y: 3.92 g      Radi Y: 39.77 cm
* * * * FINAL * * *
```

Figura 38: Sortida per pantalla de la funció calcul\_contrapesos.m

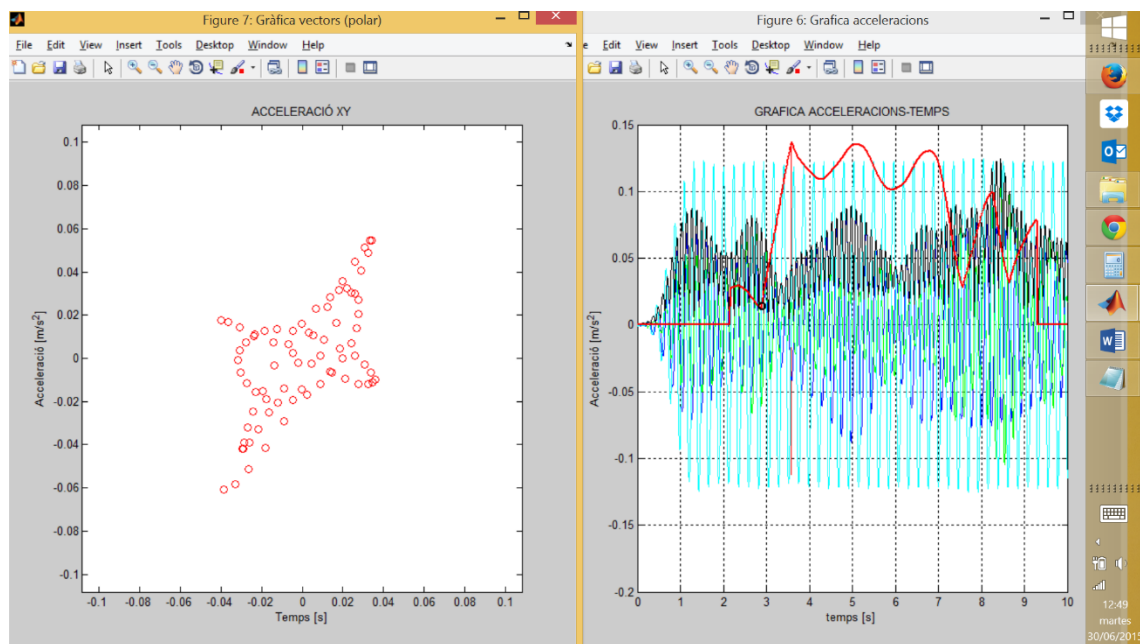


Figura 39: (Vibracions residuals) Senyal irregular i trajectòria d'acceleracions erràtica

## IV. CONCLUSIONS

El nucli del projecte ha estat el desenvolupament d'un sistema d'equilibrat de ventiladors de sostre que posteriorment pugui ser integrat en un microcontrolador.

Hem traslladat la tècnica dels equilibrats industrials als ventiladors de sostre, però hem de tenir present que no hi ha cap model d'equilibrat industrial pensat per equilibrar una màquina rotativa subjectada per un únic extrem i que manté la seva posició únicament per la força de la gravetat. Podríem dir que hem intentat “*equilibrar un pèndul*”.

El principal obstacle que hem trobat ha estat un factor de fabricació del ventilador (la diferent rigidesa en unes o altres direccions). Les forces del desequilibri “*escombren*” totes les direccions del pla XY, però en el seu recorregut troben diferents forces de reacció. Aquest fet, sumat a l'efecte pèndul que afegeix forces diferents segons la posició en cada instant, genera un moviment aparentment caòtic.

Hem hagut d'analitzar un moviment compost per aquests tres factors, intentant separar els moviments deguts a les forces ocasionades pel desequilibri de les aspes de les forces degudes a la posició en l'instant anterior i a la reacció de les subjeccions.

Partint de la base que els estàndards de qualitat d'aquest tipus de ventiladors serien baixos, hem hagut de trobar el sistema que, malgrat els factors de fabricació, aconseguís un equilibrat satisfactori.

En una primera aproximació hem aconseguit reduir les vibracions al voltant del 80%. Però per superar aquest llinar i que el sistema es pugui integrar en un microcontrolador que operi automàticament, hem hagut de construir un algoritme matemàtic capaç de seleccionar les dades òptimes per al càlcul de les correccions de masses. D'aquesta manera, el sistema redueix les vibracions per sobre del 90%, quedant unes vibracions residuals pràcticament inapreciables a ull nu.

Tot el software del sistema d'equilibrat, des de la captura de dades, passant pel filtrat de la senyal, la selecció dels valors òptims, els càlculs i la presentació de les correccions en una interfase gràfica, es pot integrar en un microcontrolador com per exemple el STM32F4. El poc hardware necessari per a l'equilibrat: un acceleròmetre digital i un sensor òptic, també pot ser gestionat per qualsevol microcontrolador de característiques similars. Els preus de mercat del hardware es situen per sota dels 50€. Aquí caldria afegir el cost del muntatge, el qual pot ser molt variable en funció de la comanda, però en tot cas, sempre quedaria lluny dels equips portàtils industrials que es poden trobar a partir de 1200€. L'operació d'equilibrat amb aquest sistema seria senzilla i fàcilment assumible amb l'ajut d'un manual d'usuari que contingui les indicacions que adjuntem a l'annex V.

D'aquesta manera es dona resposta als objectius d'autonomia, senzillesa i baix cost plantejats al principi.

Podem dir que la complexitat del projecte ha estat molt superior a la prevista en un primer moment, però precisament aquestes dificultats a l'hora de portar a la pràctica un projecte teòric han estat les més interessants des del punt de vista didàctic.

## V. CONTINUÏTAT DEL TREBALL

Abans d'integrar aquest sistema d'equilibrat en un microcontrolador recomanem millorar-ne la precisió i l'adaptabilitat a diferents models de ventiladors.

S'haurien de realitzar proves amb diferents ventiladors, i tenint en compte els paràmetres que els defineixen (distància del pal de rotació al sostre, pes real, nombre i longitud de les aspes, sistema d'ancoratge i altres factors de fabricació que considerem rellevants) mirar de trobar una equació que ens calculi automàticament una constant de correcció de linealitat (que en aquest treball hem anomenat  $K$ )

Com a mínim, provant amb un nombre considerable de ventiladors, hauríem de trobar el rang de valors que pot assolir aquesta constant i els més indicats per a cada tipologia de ventilador.

Amb l'objectiu de poder equilibrar ventiladors amb diferents nombres d'aspes, caldria realitzar el software que permetés el càlcul de la posició dels contrapesos tenint en compte la posició angular de les aspes.

Es podria simplificar el hardware del sensor òptic si es pretén implementar el sistema en un microcontrolador amb entrades analògiques que disposin de "watchdog". Això permetria prescindir del circuit de condicionament de senyal del transductor òptic, de manera que aquesta funció la realitzés el propi microcontrolador. D'aquesta manera es reduiria temps i costos de construcció i a més la detecció del pas d'aspa seria per contrast, de manera que es podria autoregular per software.

Finalment plantegem el repte d'aplicar aquest sistema d'equilibrat capturant les dades per anàlisi d'imatge. Atesa la creixent capacitat de processament dels microcontroladors, o també l'aparició al mercat de microordinadors de baix cost com per exemple la placa "*raspberrypi*", a més de càmeres de vídeo de molt baix cost, creiem que seria una opció per a fer l'equilibrat més senzill a l'operari. Ja hem vist que situant un punter làser al ventilador i projectant sobre una diana amb referències mètriques podem observar el moviment de les vibracions. A partir de l'anàlisi d'aquest moviment es podria aplicar la metodologia d'equilibrat plantejada en aquest treball.

.

## VI. RELACIÓ DE FONTS I BIBLIOGRAFIA CONSULTADA

**Análisis dinámico de máquinas rotativas por vibraciones**, Autor: Pedro Fraga López  
Edició: Universidade da Coruña. Servicio de Publicacións, 1998  
ISBN 8489694567

**Estudio de una máquina equilibradora de ejes HOFMANN modelo HL 14-1**  
Departamento de Ingeniería Mecánica, Energética y de Materiales de la Universidad pública de Navarra, autor: Juan Martín Santesteban Cizur (Projecte Final de Carrera)  
<http://academica-e.unavarra.es/bitstream/handle/2454/1861/577064.pdf>

**Tratamiento de la señal I**, autor: Francisco Clariá (Apuntes)

**Transformada Z, sistemas retroalimentados discretos y filtrado digital**, autor: Francisco Clariá (Apuntes)

**Principios de la Medición y Análisis de Vibraciones en Máquinas**, Presentado por Hofmann Mess- und Auswuchttechnik GmbH & Co. KG (Dossier de formació d'operaris d'equilibrat)

Empresa de serveis d'equilibrat: Preditec  
<http://www.preditec.com/servicios/servicio-de-equilibrado-de-rotores-in-situ-341821/>

Empresa de equilibradores electròniques i serveis d'equilibrat: Balanceo Dinámico Industrial S. A.  
[http://www.balanceodinamico.com.mx/index.php?art\\_id=5&categ=341](http://www.balanceodinamico.com.mx/index.php?art_id=5&categ=341)

Catàleg distribució ventiladors de sostres  
[http://www.bricodepot.es/nuestro-catalogo?field\\_tienda\\_asociada=40](http://www.bricodepot.es/nuestro-catalogo?field_tienda_asociada=40)

Tutorials d'equilibrat de ventiladors de sostre.  
<http://es.wikihow.com/arreglar-un-ventilador-de-techo-tambaleante>  
<https://www.youtube.com/watch?v=jWpxyoTHywA>  
<https://www.youtube.com/watch?v=yuBI10xTRxA>

**Ajuda matlab**  
<http://es.mathworks.com/help/matlab/>

**Tratamiento de datos. Ficheros con matlab**  
<http://www2.caminos.upm.es/Departamentos/matematicas/Fdistancia/PIE/matlab/temasmatlab/TEMA%208.pdf>

**Característiques físiques de les monedes d'euro**  
<http://euro.raddos.de/espanol/monedas.php>

**UM1472 - User Manual - STM32F4DISCOVERY Board**

[http://www.st.com/st-web-ui/static/active/cn/resource/technical/document/user\\_manual/DM00039084.pdf](http://www.st.com/st-web-ui/static/active/cn/resource/technical/document/user_manual/DM00039084.pdf)

**RM0090 - Reference manual - STM32F40xx advanced ARM-based 32-bit MCUs**

[http://www.st.com/web/en/resource/technical/document/reference\\_manual/DM00031020.pdf](http://www.st.com/web/en/resource/technical/document/reference_manual/DM00031020.pdf)

**DS8626 - Datasheet - STM32F407xx MCU**

[www.st.com/web/en/resource/technical/document/datasheet/DM00037051.pdf](http://www.st.com/web/en/resource/technical/document/datasheet/DM00037051.pdf)

**MPU-6000 and MPU-6050 Product Specification Revision 3.4**

<http://www.invensense.com/mems/gyro/documents/PS-MPU-6000A-00v3.4.pdf>

**MPU-6000 and MPU-6050 Register Map and Descriptions Revision 4.0**

<http://invensense.com/mems/gyro/documents/RM-MPU-6000A.pdf>

**Library 43- MPU-6050 6-axes gyro and accelerometer for STM32F4**

<http://stm32f4-discovery.com/2014/10/library-43-mpu-6050-6-axes-gyro-accelerometer-stm32f4/>

**Reflective Optical Sensor with Transistor Output (cny70)**

<http://www.vishay.com/docs/83751/cny70.pdf>



## VII. ÍNDEX DE FIGURES

Figura 1: Esquema físic i aplicació d'equilibrat estàtic d'un disc .....	11
Figura 2: Esquema físic i representació del desequilibri dinàmic en un cilindre .....	11
Figura 3: Efectes del desequilibri estàtic i dinàmic en un cilindre .....	11
Figura 4: Màquines portàtils d'equilibrat in situ.....	12
Figura 5: Màquina d'equilibrat en bancada.....	12
Figura 6: Esquerra: Senyal captada pel sensor. Dreta: Senyal filtrada .....	13
Figura 7: Interfase d'usuari per a equilibradora d'eixos en dos plans .....	13
Figura 8: Equilibrat d'un disc amb una màquina portàtil.....	14
Figura 9: Equilibrat per assaig-error d'un ventilador de sostre .....	15
Figura 10: Representació esquemàtica de la rotació d'un disc desequilibrat .....	16
Figura 11: Posició de les masses $m_1$ per neutralitzar la força $F_c$ .....	18
Figura 12: Combinacions de monedes que usarem de contrapesos .....	20
Figura 13: Sensor òptic JMN SENSOR CNY7 .....	21
Figura 14: Esquema electrònic del sensor òptic .....	22
Figura 15: Sensor òptic .....	22
Figura 16.1: Acceleròmetre analògic      Figura 17.2: Acceleròmetre digital.....	22
Figura 18: Microcontroladors: (esq.) Arduino Uno i (dr.) STM32F4 Discovery.....	23
Figura 19: Hardware necessari per a l'equilibrat de ventiladors .....	24
Figura 20: Interpretació del diagrama de flux .....	25
Figura 21: Diagrama de flux del programa inicial del microcontrolador .....	26
Figura 22: Orientació eixos acceleròmetre      Posició sensor òptic .....	27
Figura 23: Situació dels components i comprovació del funcionament del sensor òptic.....	28
Figura 24: Subjecció del ventilador amb diferent rigidesa segons la direcció .....	31
Figura 25: Espectre filtre FIR passa banda 3-4Hz ordre 197, $F_s$ 110hz. Blau: magnitud, verd: fase ...	34
Figura 26: Ventilador sense equilibrar (4 voltes anteriors al segon 3,6) .....	39
Figura 27: Ventilador sense equilibrar (4 voltes anteriors al segon 6,3) .....	39
Figura 28: Ventilador sense equilibrar (4 voltes anteriors al segon 9,1) .....	40
Figura 29: Muntatge experimental per mesurar la inclinació del ventilador en moviment.....	41
Figura 30: Ventilador amb pes de prova (4 voltes anteriors al segon 3,1) .....	42
Figura 31: Ventilador amb pes de prova (4 voltes anteriors al segon 5) .....	42
Figura 32: Ventilador amb pes de prova (4 voltes anteriors al segon 6,8) .....	43
Figura 33: Ventilador amb pes de prova (4 voltes anteriors al segon 8,7) .....	43
Figura 34: Ventilador equilibrat (entre els segons 2,5 i 3,5).....	45
Figura 35: Ventilador equilibrat (entre els segons 4,3 i 5,3).....	45
Figura 36: Ventilador equilibrat (entre els segons 6 i 7).....	46
Figura 37: Ventilador reequilibrat, acceleracions petites ( $0,04\text{m/s}^2$ ), de direcció irregular.....	52
Figura 38: Sortida per pantalla de la funció calcul_contrapesos.m.....	58
Figura 39: (Vibracions residuals) Senyal irregular i trajectòria d'acceleracions erràtica .....	59
Figura 40: Disseny de filtre FIR passa banda 3-4Hz amb FDATool.....	98

## VIII. ÍNDEX DE GRÀFIQUES

Gràfica 1: (Fs: 99Hz) Senyal captada amb el motor funcionant.....	29
Gràfica 2: (Fs: 99Hz) Ampliació de la gràfica 1.....	29
Gràfica 3: (Fs: 99Hz) Espectre de potencia de la senyal .....	30
Gràfica 4: (Fs: 99Hz) Ampliació de l'espectre de potencia gràfica 3 .....	30
Gràfica 5: (Fs: 99Hz) Engegada (segon 1) i aturada (segon 17,2) del motor.....	32
Gràfica 6: (Fs: 33Hz) Aturada del motor i reducció de la velocitat de les aspes.....	32
Gràfica 7: (Fs: 150Hz) Espectre amb forta component de senyal a 60 Hz .....	33
Gràfica 8: (Fs: 99Hz) Retard introduït per un filtre de 248 coeficients .....	35
Gràfica 9: (Fs: 110Hz) Senyal filtrada amb passabanda 2-5Hz.....	35
Gràfica 10: (Fs: 110Hz) Senyal filtrada amb passabanda 4-5Hz.....	36
Gràfica 11: (Fs: 110Hz) Acel. Z. En negre l'espectre filtrat amb passabanda 4-5Hz.....	36
Gràfica 12: Acceleracions i pas d'aspa (X blau, Y verd, Z vermell, pas d'aspa Cian) .....	37
Gràfica 13: Acceleracions X en blau, Y en verd i la seva suma vectorial en vermell.....	37
Gràfica 14: El pas per 0 de l'eix Y (verd) no concedeix amb el màxim de l'eix Y (blau) .....	38
Gràfica 15: Pèrdua de desfasament X-Y al millorar l'equilibrat.....	38
Gràfica 16: Ventilador amb pes de prova (mòdul de l'acceleració estable) .....	44
Gràfica 17: Ventilador amb pes de prova (mòdul de l'acceleració inestable) .....	44
Gràfica 18: Anàlisi de la senyal de la gràfica 19 .....	49
Gràfica 19: Ventilador en estat original. Funció "Variància_fase(t,n)" en vermell .....	49
Gràfica 20: Anàlisi de la senyal de la gràfica 21 .....	50
Gràfica 21: Ventilador equilibrat. Funció "Variància_fase(t,n)" en vermell .....	50
Gràfica 22: Anàlisi de la senyal de la gràfica 23 .....	51
Gràfica 23: Ventilador reequilibrat. Funció "Variància_fase(t,n)" en vermell .....	51
Gràfica 24: Ventilador reequilibrat. L'eix X per un període respecte al Y.....	52
Gràfica 25: Mostres òptimes i correcció amplitud. Ventilador sense equilibrar .....	55
Gràfica 26: Mostres òptimes i correcció amplitud. Ventilador equilibrat .....	55

## IX. ÍNDEX DE TAULES

Taula 1: Connexions al microcontrolador ( $\mu C$ ) .....	24
Taula 2: interfase $\mu C$ - usuari .....	26
Taula 3: Mesuraments de l'angle d'inclinació del ventilador .....	41
Taula 4: Toleràncies vibracions en funció del tipus de màquina .....	104
Taula 5: Toleràncies vibracions en funció de la freqüència de rotació.....	105
Taula 6: Límits de vibració segons la norma ISO 10816-1.....	106

## X. ANNEXES

### I. Programari microcontrolador

main.c

```

1/*****
2**
3** File      : main.c
4**
5** Equilibrador de ventiladors de sostre basat en STM32F4 i accelerometre digital MPU6050
6** Manel Vidal (Universitat de Lleida) TFG juliol 2015
7**
8** Mesurem les vibracions amb un acceleròmetre a la freqüència de mostreig Fs
9** i calculem els contrapesos per equilibrar el ventilador.
10**
11** Usarem un sensor optic per detectar cada volta del rotor
12**
13** En aquesta fase de desenvolupament del sistema d'equilibrat, el programa es limita a
capturar
14** les dades de les acceleracions, filtrar-les i enviar-les al port sèrie.
15**
16** S'incorpora una petita interfase mitjançant missatges al port sèrie o ativant i
desactivant els
17** leds de la placa STM32F4 Discovery. També s'afegeix un mode prova per saltar les
comprovacions
18** inicials (horitzontalitat de l'acceleròmetre, estabilitat del període de rotació...)
19**
20** CALCUL CONTRAPESOS (NOTA)
21** El pes que posem a les pales ha de contrarestar la força que genera l'acceleració
detectada
22** -Fcontrapes = Fventilador = m_ventilador x Accel_ventilador
23** On Fcontrapes = m_contrapes x Accel_normal_contrapes = m_contrapes · R · w^2
24** Amb 4 pales, R·cos(Fi) = Rx (distància a la que he posar el pes sobre les aspes de l'eix
X.
25** R·sin(Fi) = Ry (ídem per l'eix Y. El valor positiu o negatiu indica l'aspa)
26** EQUACIONS: Rx = (mv·av/m_contrapes·w)· cos(180+Fi); Ry = (mv·av/m_contrapes·w)·
sin(180+Fi);
27** Algoritme: Buscar pes suficient per Rx i Ry màxims (la longitud de l'aspa)
28** Per aquest pes, trobar Rx i Ry
29**
30**
31** The USART2 Configuration for the debug is:
32**
33** - BaudRate = 115200 baud
34** - Word Length = 8 Bits
35** - One Stop Bit
36** - No parity
37** - Hardware flow control disabled (RTS and CTS signals)
38**
39**
40*****/
41
42// Includes
43#include "..\Discovery\defines.h"
44
45#define VOLTES 20 // Voltes sobre les que calculem les variances i mitjanes; maxim 256
(uint8_t)
46#define MOSTRES 1100 //longitud dels vectors de mostreig
47#define FS 110 //Freqüència de mostreig
48#define LIM_VAR_MODUL 200 //Límit de la variança del mòdul
49#define LIM_VAR_ANGLE 200 //Límit de la variança de angle
50#define LIMIT_ACCELEROMETRE_HORIZTONTAL 500 // Màxima acceleració admesa per inclinació de
l'acceleròmetre amb ventilador parat
51
52#define RADII_PES_PROVA 0.5 // Distància R del centre del ventilador a on col·loquem el pes
de prova
53

```

```
54
55//Variables Externes
56extern uint8_t mode_prova; // L'activa i desactiva el polsador blau
57extern uint8_t pas; // Indicarà la activació de la interrupció pas d'aspa (quan la detecta
    el sensor òptic)
58uint8_t OK = RESET; // variable per comprovacions
59
60// Global variable
61extern uint8_t doNewSample; //Indica si cal fer lectura d'acceleracions, l'activa el
    Timer_6 a Fs
62extern uint32_t nTIM6OF; //Contador de flag del timer 6
63uint32_t retard= (int)NUM_TAPS*1.2; //retard del filtre en nombre de mostres. 1,2 * núm.
    coeficients filtre
64uint32_t mostres=MOSTRES; // mostres totals de senyal capturada
65uint32_t i; uint32_t j; uint32_t k; uint32_t cont; //Variables per bucles
66
67// Variable que defineix l'estat del procés
68uint8_t estat = 0;
69
70// Vectors senyal acceleració i sensor òptic
71static float accel_X[MOSTRES];
72static float accel_Y[MOSTRES];
73static float accel_Z[MOSTRES];
74static float sensor_optic[MOSTRES];
75
76// Vectors senyal Filtrada acceleració i sensor òptic
77static float accel_Xf[MOSTRES];
78static float accel_Yf[MOSTRES];
79static float accel_Zf[MOSTRES];
80static float sensor_opticf[MOSTRES];
81static float accel_XYf[MOSTRES]; // Suma vectorial de X+Y
82
83// Variables per valor inicial de l'acceleració (per determinar inclinació acceleròmetre)
84float accel_Xo;
85float accel_Yo;
86float accel_Zo;
87
88//Variables per freqüència, període i fase
89int Fs = FS; //Freqüència de mostreig
90float freq = 0; //Freqüència de rotació
91float w; //velocitat angular
92float rpm; //r.p.m.
93float T[VOLTES+1]; //vector per emmagatzemar els períodes
94float modul;
95uint32_t posicio_modul_maxim[VOLTES+1];
96uint32_t posicio_pas_aspa[VOLTES+1];
97float angle;
98float modul_inicial = -1; // Inicialitzem amb valor negatiu per saber que encara no l'hem
    usat.
99float angle_inicial;
100float radi_aspa_cm; //radi màxim de l'aspa (paràmetre del ventilador)
101float radi_pes_1=0; // Distància al centre de rotació [cm] on es col·locarà el pes en l'eix X
    (aspa amb reflectant)
102float radi_pes_2=0; // " .... on es col·locarà el pes en l'eix Y
103
104
105#define NOMBRE_PESOS 8 // Definim el nombre de pesos de que disposarem
106// Definim el joc de pesos que disposarem. El valor 1kg indicara fora de rang
107float massa[NOMBRE_PESOS] = {0.00230, 0.00306, 0.00392, 0.00410, 0.00574, 0.0078, 0.0085,
    1};
108float massa_pes_prova[1]={0.00392}; //Variable vector per si més endavant usem altres pesos
109
```

```

110/* Pesos de les monedes
111% Seqüència: posició vector, pes [kg], monedes:
112% 1 0.00230 --> 1 x 1 cèntim
113% 2 0.00306 --> 1 x 2 cèntims
114% 3 0.00392 --> 1 x 5 cèntims
115% 4 0.00410 --> 1 x 10 cèntim
116% 5 0.00574 --> 1 x 20 cèntims
117% 6 0.00780 --> 1 x 50 cèntims
118% 7 0.00850 --> 1 x 2 euros
119% 8 --> ERROR!
120%
121% PES DE PROVA
122% 1 0.00392 --> 1 x 5 cèntims
123% */
124
125//Variable que posem a 1 a cada interrupció del sensor òptic
126//(Definida al EXTI_1_PA1.C com a externa)
127//int pas=0;
128
129
130
131
132
133
134// Main program
135int main(void) {
136
137    // Configures SysTick for delay functions
138    Delay_Init();
139    // Initialize all LEDs
140    LEDInitAll();
141    // USART2 Initialization
142    USARTx_Init(USART_MODE_IRQ);
143    // Inicialitzar interrupció Sensor optic
144    EXTI_1_PA1_Init();
145    // Inicialitzar User Button (Per activar o desactivar el mode_prova)
146    PushButtonInit(BUTTON_USER, BUTTON_MODE_EXTI);
147
148    if (mode_prova == SET){LEDOn(LED_ORANGE);}
149
150    /* Initialize MPU6050 sensor 0, address = 0xD0, AD0 pin on sensor is low */
151    TM_MPU6050_t MPU6050_Data0;
152    if (TM_MPU6050_Init(&MPU6050_Data0, TM_MPU6050_Device_0, TM_MPU6050_Accelerometer_2G,
        TM_MPU6050_Gyroscope_250s) == TM_MPU6050_Result_0k) {
153        /* Display message to user */
154        printf("Accelerometer is ready to use!\r");
155    }else{
156        printf("\r\rERROR COMUNICACIO Accelerometer 6050\rRevisi connexions\r\r");
157        alarma(1);
158    }
159
160    // Confiure and start the Timer 6 with an interrupt update at int freq. [Hz]
161    // Com l'usem amb diferents freqüències, el configurarem allà on l'utilitzem amb Fs
    corresponent.
162    //TIM6_Config(Fs);
163
164    mode_prova=RESET;
165
166
167
168    while (1) {
169        Delay_ms(3000);//Per donar temps a posar mode prova=SET (Push button --> Led

```

```
taronja encés)
170
171
172 while (estat==0) { //Per si volem accedir a un menu de configuració de paràmetres
173     //...
174     radi_aspa_cm=50; //paràmetre del ventilador utilitzat per les proves
175
176     estat=1;
177 }
178
179 while (estat==1) { //Per si volem presentar els paràmetres del ventilador)
180     //Esperem confirmació o canvi paràmetres
181     //...
182     estat=2;
183 }
184
185 while (estat==2) { //Mesurar posició inicial acceleròmetre
186     //Assegurar-se que el ventilador esta en repòs
187
188     //En mode prova saltem aquest estat i el següent
189     if (mode_prova == SET){
190         T[0] = 0.2255; freq = 1/T[0];
191         TIM6_Config(Fs); doNewSample=RESET; pas = RESET;
192         estat=4;
193         OK=SET;
194     }else{OK=RESET;}
195
196     // Comprobem si el ventilador està aturat i si l'acceleròmetre està horitzontal
197     while (OK == RESET){
198         for (i=0; i < VOLTES; i++) {
199             TM_MPU6050_ReadAccelerometer(&MPU6050_Data0);
200             accel_X[i] = (float)MPU6050_Data0.Accelerometer_X;
201             accel_Y[i] = (float)MPU6050_Data0.Accelerometer_Y;
202             accel_Z[i] = (float)MPU6050_Data0.Accelerometer_Z;
203             Delay_ms(50);
204         }
205         if (varianza(&accel_X[0], VOLTES, 4000) + varianza(&accel_Y[0], VOLTES,
206             4000) + varianza(&accel_Z[0], VOLTES, 8000) > 0){
207             printf("Detecta moviment, aturi ventilador\r");
208         }else{
209             printf("Mesurada posicio inicial (aturat)\r");
210             accel_Xo = mitja(&accel_X[0], VOLTES);
211             accel_Yo = mitja(&accel_Y[0], VOLTES);
212             accel_Zo = mitja(&accel_Z[0], VOLTES);
213             printf("Xo: %.2f \tYo: %.2f \tZo: %.2f \r \r", accel_Xo, accel_Yo,
214                 accel_Zo);
215             if (powf(accel_Xo,2) < powf(LIMIT_ACCELEROMETRE_HORIZTONTAL,2) &&
216                 powf(accel_Yo,2) < powf(LIMIT_ACCELEROMETRE_HORIZTONTAL,2) ){
217                 printf("\rAccelerometre Ok\r");
218                 OK=SET;
219             }else{
220                 printf("\rAccelerometre INCLINAT\rLimit acceleracio per inclinacio
221                 = %.2f\r", (float)LIMIT_ACCELEROMETRE_HORIZTONTAL);
222                 OK=RESET;
223             }
224         }
225     }
226     if (mode_prova != SET && OK == SET){estat=3;}
227 }
228
229 while (estat==3) { //Mesurar període i freqüència amb sensor òptic (per seleccionar
```



```

    el filtre)
227     printf("Posi en marxa el ventilador\r");
228     // Aprofitem el timer del mostreig (li apliquem 1kHz)
229     TIM6_Config(1000); //1000 Hz és la Fs màx. amb el preescaler de 8400
230     pas = RESET;
231     while (pas==RESET){Delay_us(1);} //esperem el primer flanc de pas d'aspa
232     TIM_SetCounter(TIM6, 0); nTIM6OF=0; //Reiniciem el timer
233     OK=RESET;
234     while (OK == RESET){
235         for (i=0; i < VOLTES; i++){
236             pas=RESET;
237             while (pas==RESET){Delay_us(1);}
238             T[i]= ((float)TIM_GetCounter(TIM6)*0.001f/10.0f) +
((float)nTIM6OF*0.001f);
239             TIM_SetCounter(TIM6, 0); nTIM6OF=0;
240             //printf("\tPeriode: %.9f \r \r", T[i]);
241         }
242         if (varianza(&T[0], VOLTES, 0.0000001) > 0){
243             printf("\tVARIANZA x 1000: %6f \r", 1000*varianza(&T[0], VOLTES,
0.0000001));
244         }else{
245             printf("VAR_Periode OK: %9f ms \r", varianza(&T[0], VOLTES,
0.0000001));
246             printf("Mesurat periode (estacionari): ");
247             T[0] = mitja(&T[0], VOLTES);
248             freq = 1/T[0];
249             printf("%.4f \t, frecuencia: %.3f\t Fs: %d Hz\r", T[0], freq, Fs);
250             if (Fs > (int)10*freq){printf("Fs OK!\r\r");}else{printf("Fs
INSUFICIENT!\r\r");Delay_ms(5000);}
251             OK=SET;
252         }
253     }
254     // Disable Timer 6
255     TIM_Cmd(TIM6, DISABLE);
256
257     estat=4;
258 }
259
260
261 while (estat==4) { //Mostrejar i filtrar
262     if (mode_prova == SET){
263         mode_prova=RESET;
264         LEDOff(LED_ORANGE);
265         printf("\rBOTO BLAU PER INICIAR LECTURA ACCELERACIONS\r");
266         while(mode_prova==RESET){Delay_us(1);}
267         printf("\rMode Prova= %d\r", mode_prova);
268         Delay_ms(50);
269     }
270
271
272     //Llegim acceleracions a cada flag del timer fins al nombre de MOSTRES
273     printf("\r\rLlegint acceleracions\r");
274     i=0;
275     // Reiniciem les variables de mostreig i pas d'aspa. Configurem el timer a Fs
276     TIM6_Config(Fs); doNewSample=RESET; pas = RESET;
277     TIM_SetCounter(TIM6, 0); nTIM6OF=0; //Reiniciem el timer
278     while (i < MOSTRES){
279         if (doNewSample==SET){
280             TM_MPU6050_ReadAccelerometer(&MPU6050_Data0);
281             accel_X[i] = (float)MPU6050_Data0.Accelerometer_X;
282             accel_Y[i] = (float)MPU6050_Data0.Accelerometer_Y;
283             accel_Z[i] = (float)MPU6050_Data0.Accelerometer_Z;

```

```
284         sensor_optic[i] = (float)pas*1000;
285         doNewSample=RESET;
286         pas = RESET;
287         i++;
288     }
289     Delay_us(1); //És necessari posar alguna instrucció dins un while. Sinó es
penja.
290 }
291
292 if (mode_prova == SET){
293     mode_prova=RESET;
294     LEDOff(LED_ORANGE);
295     printf("\rSenyal acceleracio guardada (NO FILTRADA)\rBOTO BLAU PER INICIAR
ENVIAMENT A PORT SERIE\r");
296     while(mode_prova==RESET){Delay_us(1);};
297     Delay_ms(50);
298     for (i=0; i < MOSTRES; i++){
299         //Enviem al port sèrie
300         printf("%.0f\n", accel_X[i]);
301         printf("%.0f\n", accel_Y[i]);
302         printf("%.0f\n", accel_Z[i]);
303         printf("%.0f\n", sensor_optic[i]);
304         Delay_ms(20); //Temps d'espera per no saturar el buffer del port sèrie
305     }
306 }
307
308
309 // FILTRAT X, Y, Z i PAS d'aspa
310 printf("\r\nIniciem filtrat\r");
311 TIM_SetCounter(TIM6, 0); nTIM6OF=0; //Reiniciem el timer per calcular el temps
de filtrat
312 // tipus de variables de la funció fir_fs110
313 //uint8_t fir_fs110(float freq, uint32_t mostres, float32_t *inputF32,
float32_t *outputF32);
314 fir_fs110(freq, mostres, &accel_X[0], &accel_Xf[0]);
315 fir_fs110(freq, mostres, &accel_Y[0], &accel_Yf[0]);
316 fir_fs110(freq, mostres, &accel_Z[0], &accel_Zf[0]);
317 fir_fs110(freq, mostres, &sensor_optic[0], &sensor_opticf[0]);
318
319 printf("\rFI FILTRAT: Temps: %f\r", (float)TIM_GetCounter(TIM6)*0.001f/10.0f) +
(float)nTIM6OF*0.001f;
320
321 if (mode_prova == SET){
322     mode_prova=RESET;
323     LEDOff(LED_ORANGE);
324     printf("\rSenyal acceleracio FILTRADA\rBOTO BLAU PER INICIAR ENVIAMENT A
PORT SERIE\r");
325     while(mode_prova==RESET){Delay_us(1);};
326     Delay_ms(50);
327     //ENVIAMENT AL PORT SERIE
328     for (i=0; i<MOSTRES; i++){
329         printf("%f\n", accel_Xf[i]);
330         printf("%f\n", accel_Yf[i]);
331         printf("%f\n", accel_Zf[i]);
332         printf("%f\n", sensor_opticf[i]);
333         Delay_ms(20);
334     }
335 }
336
337 estat=5;
338 }
339
```



```

340/** En la fase de desenvolupament els següents càlculs es realitzen a l'ordinador amb
    Matlab
341 * Abans de fer els càlculs, amb un algoritme creat amb Matlab es localitzen les mostres
    òptimes
342 * per millorar la precisió. Un cop provat l'algoritme es pot afegir als passos següents
343 * del programa del microcontrolador.
344**/
345
346     while (estat==5) { //Extreure mòdul, angle i període de la senyal
347
348         // Començarem a buscar a la senyal a partir del període de retard del filtre
        (ordre) + 20% per estabilització (1.2 x NUM_TAPS)
349         // SUMA VECTORIAL X + Y (Z no aporta acceleració significativa sobre les
        vibracions en el pla XY)
350         for (i = retard-2; i < MOSTRES; i++){ //recorro el vector a partir del retard.
            Agafo 2 mostres anteriors per les comparacions
351             accel_XYf[i]=sqrtf(powf(accel_Xf[i],2)+powf(accel_Yf[i],2));
352         }
353
354         // Busquem el mòdul AMPLITUD (En un tram de senyal amb poca varianza)
355         //Defeneixo el temps d'integració a partir del nombre de voltes i el període
        --> temps_integracio = VOLTES * T[0];
356         //mostres que integraré = temps_integracio * Fs
357         uint16_t mostres_temps_integracio = floor(VOLTES * T[0] * Fs);
358         printf("\rMostres: %d\tRetard: %d\t Mostres temps integracio: %d\rMostres
        sobrants: %d", mostres, retard, mostres_temps_integracio,
        mostres-mostres_temps_integracio-retard);
359
360         // Comprovo la variança del mòdul (X+Y) en aquest tram i si cal desplaço el
        tram
361         printf("\rComprovant variança acceleracio\r");
362         while ( (varianza(&accel_XYf[retard], mostres_temps_integracio, 1000) > 0) &&
        (mostres-mostres_temps_integracio-retard > 1) ){
363             printf("\rMòdul de l'acceleració poc estable\r");
364             printf("Darrera variança: %f \r", varianza(&accel_XYf[retard],
        mostres_temps_integracio, 1000));
365             retard = retard + 10; //Desplaço 10 mostres
366             printf("\rMostres: %d\tRetard: %d\t Mostres temps integracio: %d\rMostres
        sobrants: %d", mostres, retard, mostres_temps_integracio,
        mostres-mostres_temps_integracio-retard);
367         }
368
369         //Comprovo que disposi de prou senyal (suficients mostres) pels càlculs
370         if (retard > mostres-mostres_temps_integracio-1){
371             printf("\rVarianza excesiva. Torni a mesurar acceleracions\rREINICI EL
        PROCES");
372             alarma(estat);
373         }
374
375         //Integrem el mòdul X+Y per calcular la mitjana
376         modul=0;
377         float modul_maxim_XYf = 0, modul_maxim_Xf = 0, modul_maxim_Yf = 0;
378         uint32_t posicio_modul_maxim_XYf, posicio_modul_maxim_Xf,
        posicio_modul_maxim_Yf;
379
380         for (i=retard; i < retard + mostres_temps_integracio; i++){
381             modul = modul + accel_XYf[i];
382             // Guardo les posicions on tinc els mòduls maxims per comprovar desviacions
383             if (accel_XYf[i] > modul_maxim_XYf){modul_maxim_XYf =
        accel_XYf[i]; posicio_modul_maxim_XYf = i;}
384             if (accel_Xf[i] > modul_maxim_Xf){modul_maxim_Xf = accel_Xf[i];
        posicio_modul_maxim_Xf = i;}

```

```
385         if (accel_Yf[i] > modul_maxim_Xf){modul_maxim_Yf =  
    accel_Yf[i];     posicio_modul_maxim_Yf = i;}  
386     }  
387     modul = modul / mostres_temps_integracio;  
388     // En unitats del S.I.[m/s^2]. El sensor dóna +/- 2G (G=9,81[m/s^2]) amb +/-  
    2^16 valors  
389     modul = 2*9.81*modul/32768;  
390     printf("\rModul mitja Acceleracio: %f", modul);  
391     modul_maxim_XYf = 2*9.81*modul_maxim_XYf/32768;  
392     printf("\rModul maxim %f, Modul Xf %f, Modul Yf %f, posicio (s) %f",  
    modul_maxim_XYf, accel_Xf[posicio_modul_maxim_XYf], accel_Yf[posicio_modul_maxim_XYf],  
    1.0*posicio_modul_maxim_XYf/Fs);  
393     printf("\rAngle maxim acceleracio (º) %f",  
    atan2(accel_Yf[posicio_modul_maxim_XYf], accel_Xf[posicio_modul_maxim_XYf])*180/PI );  
394     modul_maxim_Xf = 2*9.81*modul_maxim_Xf/32768;  
395     printf("\rModul maxim Xf %f, Modul Yf %f, posicio (s) %f", modul_maxim_Xf,  
    accel_Yf[posicio_modul_maxim_Xf], 1.0*posicio_modul_maxim_Xf/Fs);  
396     printf("\rAngle maxim Xf (º) %f", atan2(accel_Yf[posicio_modul_maxim_Xf],  
    accel_Xf[posicio_modul_maxim_Xf])*180/PI );  
397     modul_maxim_Yf = 2*9.81*modul_maxim_Yf/32768;  
398     printf("\rModul maxim Yf %f, Modul Xf %f, posicio (s) %f", modul_maxim_Yf,  
    accel_Xf[posicio_modul_maxim_Yf], 1.0*posicio_modul_maxim_Yf/Fs);  
399     printf("\rAngle maxim Yf (º) %f", atan2(accel_Yf[posicio_modul_maxim_Yf],  
    accel_Xf[posicio_modul_maxim_Yf])*180/PI );  
400  
401  
402     //Extreure l'angle mitjà de l'acceleració  
403  
404     //Busquem l'angle de l'acceleració quan detecta el sensor òptic  
405     //Primer Busco els passos per Zero (de negatiu a positiu) del sensor òptic. (més  
    precís que els màxims)  
406     j=0;  
407     for (i=retard; i < retard + mostres_temps_integracio; i++){  
408         if ( (sensor_opticf[i+1] > 0) && (sensor_opticf[i] < 0) ){  
409             // busco la mostra més proera a Zero  
410             if (sensor_opticf[i+1] > (0 - sensor_opticf[i])){  
411                 //Assigno les posicions  
412                 posicio_pas_aspa[j] = i;  
413             }else{  
414                 posicio_pas_aspa[j] = i+1;  
415             }  
416             j=j+1;  
417         }  
418     }  
419     j=j-1; //j=nombre de passos per zero de negatiu a positiu  
420  
421     //Calculo el periode del ventilador en aquest tram  
422     T[0] = (float)(posicio_pas_aspa[j]- posicio_pas_aspa[0]) / ((j)*Fs);  
423  
424     //Li sumo les mostres corresponents a 1/4 del periode (90º)  
425     //Serà la mostra on tinc el màxim del sensor òptic.  
426     // Les mostres amb què calcular l'angle d'acceleració amb X i Y  
427     for (i=0; i<j;i++){  
428         posicio_pas_aspa[i] = posicio_pas_aspa[i] + (uint16_t)(T[0]*Fs/4);  
429     }  
430  
431     //Trobar l'angle de l'acceleració quan el sensor òptic detecta  
432     angle = 0;  
433     for (i=0; i < j;i++){  
434         angle = angle + atan2(accel_Yf[posicio_pas_aspa[i]],  
    accel_Xf[posicio_pas_aspa[i]]);  
435     }
```

```

436         angle = angle / j;
437
438         printf("\rACCELERACIO\tMODUL: %f m/s^2\tAngle: %f °\tPeriode: %f\r", modul,
angle*180/PI, T[0]);
439         estat=6;
440     }
441
442
443
444     while (estat==6) {//Col·locar pes prova i repetir el procés
445
446         //El primer cop guardem modul i angle inicials
447         if (modul_inicial < 0){ // modul_inicial = - 1 el primer cop que entra al estat
7
448             modul_inicial = modul;
449             angle_inicial = angle;
450             // Demanar que es col·loqui el pes de prova a distància R a la pala del
reflectant.
451             printf("\rCol·loqui el pes de prova: Massa %.2f g, a l'extrem de l'aspa amb
reflectant", massa_pes_prova[0]*1000);
452             // i repetim el procés amb el contrapés
453             estat = 4;
454         }else{
455             printf("\rACCELERACIO AMB CONTRAPES\r\tMODUL: %f m/s^2\tAngle: %f
°\tPeriode: %f\r", modul, angle*180/PI, T[0]);
456             Delay_ms(5000);
457             estat = 7;
458         }
459     }
460 }
461
462     while (estat==7) {//Calcular correccions
463
464         //Calcul de la massa equivalent del ventilador
465         // (en base al canvi d'acceleracions ocasionat pel pes de prova)
466         rpm = 60/T[0];
467         w = 2*PI/T[0];
468
469         // Variació de l'acceleració provocada pel pes de prova
470         float AppY;
471         //Eix Y (Si l'orientació de l'exi X el sensor òptic és correcta hauria de donar
0
472         // Calculem la variació per poder fer comprovacions
473         AppY = (modul * sin(angle)) - (modul_inicial * sin(angle_inicial));
474
475         float AppX;
476         AppX = (modul * cos(angle)) - (modul_inicial * cos(angle_inicial));
477
478         // Força del pes de prova
479         float Fpp;
480         Fpp = massa_pes_prova[0] * RADI_PES_PROVA * powf(w,2);
481
482         // Massa equivalent del ventilador
483         float Mventilador;
484         Mventilador = Fpp / sqrtf(powf(AppX,2)+powf(AppY,2));
485         printf("\rMassa equivalent ventilador: %.2f kg\r", Mventilador);
486
487         // Càlcul contrapesos
488         // Massa: Augmentem la massa del contrapés fins que sigui sufient a un Radi <=
a longitud aspa
489         i=0;
490         //Busquem un pes suficient pel l'exi X

```

```
491         k = (Mventilador * modul_inicial * cos(PI+angle_inicial)) / (radi_aspa_cm *  
0.01 * powf(w,2));  
492         if (k<0){k=-k;} //Valor absolut de k  
493         while (k > massa[i]){i++;} //Augmentem i per l'eix x  
494         k = (Mventilador * modul_inicial * sin(PI+angle_inicial)) / (radi_aspa_cm *  
0.01 * powf(w,2));  
495         if (k<0){k=-k;}  
496         //Si cal, busquem que el pes també sigui suficient pel l'eix Y  
497         while (k > massa[i]){i++;}  
498         // Si no dispo de prous pesos per l'acceleració detectada, indico error  
499         if (i > NOMBRE_PESOS-2){ // (Si i se surt del vector)  
500             alarma(estat);  
501             } // Assenyala l'error dona el valor 10 a i.  
502  
503         // Posició  
504         radi_pes_1 = 100*(Mventilador * modul_inicial * cos(PI+angle_inicial)) /  
(massa[i] * powf(w,2));  
505         radi_pes_2 = 100*(Mventilador * modul_inicial * sin(PI+angle_inicial)) /  
(massa[i] * powf(w,2));  
506  
507  
508  
509         // ENVIAMENT DE DADES AL PORT SERIE  
510         if (mode_prova == RESET){  
511             printf("r.p.m.: %04.0f, Hz: %2.2f\r", rpm, rpm/60);  
512             printf("Modul acceleració: %2.2f\r", modul_inicial);  
513             printf("Angle acceleració: %2.2f\r", angle_inicial);  
514             printf("Joc de pesos: %d, (%.1f grams)\r", i, massa[i]*1000);  
515             if (i > NOMBRE_PESOS - 2){  
516                 printf("ERROR O DESEQUILIBRAT EXCESSIU!!\r");  
517             }  
518             printf("Distancia del pes al centre en l'eix de deteccio de pas: %+02f  
cm\r", radi_pes_1);  
519             printf("Distancia del pes al centre en l'altre eix: %+02f cm\r\r",  
radi_pes_2);  
520  
521         }else{  
522             // enviar valors a port serie per interpretar Labview  
523             // rpm_x.xx+y.yy+px+py [rpm_ (revolucions per minut 4 digits), x.xx  
(acceleració x), y.yy (acceleració y), pe (núm del 00 al 10 que indica joc de pesos), px  
(distància del pes al centre en cm pala x), py (ídem pala y)  
524             printf("%04.0f", rpm);  
525             printf("%+1.2f", modul_inicial);  
526             printf("%+1.2f", angle_inicial);  
527             printf("%02d", i);  
528             printf("%+02f", radi_pes_1);  
529             printf("%+02f\n", radi_pes_2);  
530         }  
531     }  
532  
533     alarma(8); //Parpalleig del led blau 8 cops i pausa de 3s. Indica fi del procés  
534 } //Fi while(1)  
535 } //Fi main  
536 }
```

## defines.h

```
1
2
3#include "arm_math.h"
4#include "stm32f4xx.h"
5#include ".\Discovery\led.h"
6#include ".\Discovery\pushbutton.h"
7#include ".\SysTick\delay.h"
8#include ".\USART\usart.h"
9#include ".\Filters\fir_fs110.h"
10#include ".\MPU6050\MPU_6050\tm_stm32f4_mpu6050.h"
11#include ".\MPU6050\EXTI_1_PA1.h"
12
13
14
15/* Put your global defines for all libraries here used in your project */
16//DEFINES
17#define PI 3.14159265358979f
18
19
20// External function prototypes (defined in tiny_printf.c)
21extern int printf(const char *fmt, ...);
22//extern int sprintf(char* str, const char *fmt, ...);
23
24
25//FUNCTION
26//Configuració del Timer (li entrarem per variable al Freqüència de mostreig
27void TIM6_Config(int);
28//Llegir Acceleracions
29//void ReadAcceleracions(void);
30//variança (Retornarà 0 si es correcta)
31float varianza(float *vector, uint8_t n, float limit); //Apuntador del vector, num de
    mostres, valor limit acceptable
32//Mitjana
33float mitja(float *vector, uint8_t n);
34//Visualitzem amb parpalleig del Led Blue l'estat on hem tingut l'error
35void alarma(uint8_t numero);
36
```

defines.c

```
1
2// Funcions i configuracions creades per al projecte
3
4#include "defines.h"
5
6// Global variable
7uint8_t doNewSample = RESET; //Indica si cal fer lectura d'acceleracions, l'activa el
    Timer_6 a Fs
8uint32_t nTIM6OF = 0; //Contador de flag del timer 6
9
10
11//Llegir Acceleracions MPU6050
12//void ReadAcceleracions(void){}
13
14//Funció per calcular variances
15float varianza(float *vector, uint8_t n, float limit){ //Apuntador del vector, num de
    mostres, valor límit acceptable
16    float varian=0;
17    float mitjana;
18    mitjana = mitja(&vector[0],n);
19    //printf("Mitjana: %d", mitjana);
20    int i;
21    for (i=0; i<n;i++){
22        varian = varian + powf(mitjana - vector[i],2); //Si faig ^2 no va bé
23        //printf("bucle funcio varianza\r");
24        //printf("i= %d\tvarian= %f\t vector[i]= %f\r",i, varian, vector[i] );
25        //Delay_ms(500);
26    }
27    //Delay_ms(500);
28    varian = varian/(n*1.0);
29    if (varian < limit){
30        return 0;
31    }else return varian;
32}
33
34
35//Funció per calcular mitjanes
36float mitja(float *vector, uint8_t n){
37    int i;
38    float mitjana = 0;
39    for (i=0; i<n;i++){
40        mitjana = mitjana + vector[i];
41    }
42    mitjana = mitjana/n;
43    return mitjana;
44}
45
46
47void alarma(uint8_t numero){
48    int i;
49    while(1){
50        LEDOff(LED_BLUE);
51        for (i=0;i<2*numero;i++){
52            LEDToggle(LED_BLUE);
53            Delay_ms(500);
54        }
55        Delay_ms(3000);
56    }
57}
58
59/**
60 * @note    PCLK1 = HCLK/4
61 *          TIM6CLK = (PCLK1)x2 = (168 MHz/4)x2 = 84 MHz
62 *
63 *          so CLK_INT = TIM6CLK/((Prescaler+1) * Clock_Division)
64 *          TIM6 update time = (1/CLK_INT) * (Period+1)
65 *          TIM6 update freq = CLK_INT / (Period+1)
66 *
67 *          Own Config:
68 *          -----
```

```

69 *      Precaler = 19999
70 *      Period = 41999 => 42000 ticks (from 0 to 41999)
71 *      CLK_INT = 84MHz/((19999+1) * 1) = 4.20KHz
72 *      TIM6 update freq = 4.20KHz / (41999 + 1) = 10 Hz => 100 ms
73 *
74 *      GENERA UNA INTERRUPTIÓ CADA 100ms
75 */
76 void TIM6_Config(int Fs) {
77     int T_ms=1000/Fs; //T en ms i Fs en Hz
78
79     TIM_TimeBaseInitTypeDef TIM_TimeBaseStructure;
80     NVIC_InitTypeDef NVIC_InitStructure;
81
82     // TIM6 clock enable
83     RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM6, ENABLE);
84
85     // Configure the NVIC Priority Grouping to 2:
86     // - 2 bits for preemption priority
87     // - 2 bits for subpriority
88     NVIC_PriorityGroupConfig(NVIC_PriorityGroup_2);
89
90     // NVIC Configuration
91     // Enable the TIM6 global Interrupt with lower priority (with NVIC_PriorityGroup_2)
92     NVIC_InitStructure.NVIC_IRQChannel = TIM6_DAC_IRQn;
93     NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 3;
94     NVIC_InitStructure.NVIC_IRQChannelSubPriority = 2; //Li donem més prioritat que a la
    interrupció del sensor òptic
95     NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;
96     NVIC_Init(&NVIC_InitStructure);
97
98     // Time base configuration
99     TIM_TimeBaseStructure.TIM_Period = (10*T_ms)-1; //0,1ms x T
100    TIM_TimeBaseStructure.TIM_Prescaler = 8400-1; //Passo de 84MHz a 10kHz
101    TIM_TimeBaseStructure.TIM_ClockDivision = TIM_CKD_DIV1;
102    TIM_TimeBaseStructure.TIM_CounterMode = TIM_CounterMode_Up;
103    TIM_TimeBaseInit(TIM6, &TIM_TimeBaseStructure);
104
105    // Enable Interruption of peripheral TIM6 Update
106    TIM_ITConfig(TIM6, TIM_IT_Update, ENABLE);
107
108    // Start counting time
109    TIM_SetCounter(TIM6, 0);
110
111    // Enable Timer 6
112    TIM_Cmd(TIM6, ENABLE);
113}
114
115
116
117
118/**
119 * @brief This function handles TIM6 global interrupt request.
120 * @param None
121 * @retval None
122 */
123 void TIM6_DAC_IRQHandler(void) {
124
125     // Check if TIM6 Interrupt update is asserted
126     if (TIM_GetITStatus(TIM6, TIM_IT_Update) != RESET) {
127
128         nTIM6OF++;
129         doNewSample=SET;
130         LEDToggle(LED_GREEN);
131
132         // We need to clear line pending bit manually
133         TIM_ClearITPendingBit(TIM6, TIM_IT_Update);
134     }
135}

```



pushbutton.c

```
1#include "pushbutton.h"
2
3/* Variables -----*/
4GPIO_TypeDef*  BUTTON_PORT[BUTTONn] = {USER_BUTTON_GPIO_PORT };
5const uint16_t BUTTON_PIN[BUTTONn] = {USER_BUTTON_PIN };
6const uint32_t BUTTON_CLK[BUTTONn] = {USER_BUTTON_GPIO_CLK };
7const uint16_t BUTTON_EXTI_LINE[BUTTONn] = {USER_BUTTON_EXTI_LINE };
8const uint8_t  BUTTON_PORT_SOURCE[BUTTONn] = {USER_BUTTON_EXTI_PORT_SOURCE};
9const uint8_t  BUTTON_PIN_SOURCE[BUTTONn] = {USER_BUTTON_EXTI_PIN_SOURCE };
10const uint8_t  BUTTON_IRQn[BUTTONn] = {USER_BUTTON_EXTI_IRQn };
11
12uint8_t mode_prova;
13
14// Initialization of PushButton
15void PushButtonInit(Button_TypeDef Button, ButtonMode_TypeDef Button_Mode) {
16    GPIO_InitTypeDef GPIO_InitStructure;
17    EXTI_InitTypeDef EXTI_InitStructure;
18    NVIC_InitTypeDef NVIC_InitStructure;
19
20    /* Enable the BUTTON Clock */
21    RCC_AHB1PeriphClockCmd(BUTTON_CLK[Button], ENABLE);
22
23    /* Configure Button pin as input */
24    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_IN;
25    GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_NOPULL;
26    GPIO_InitStructure.GPIO_Pin = BUTTON_PIN[Button];
27    GPIO_Init(BUTTON_PORT[Button], &GPIO_InitStructure);
28
29    if (Button_Mode == BUTTON_MODE_EXTI) {
30        /* Enable the SYSCFG Clock */
31        RCC_APB2PeriphClockCmd(RCC_APB2Periph_SYSCFG, ENABLE);
32
33        /* Connect Button EXTI Line to Button GPIO Pin */
34        SYSCFG_EXTILineConfig(BUTTON_PORT_SOURCE[Button], BUTTON_PIN_SOURCE[Button]);
35
36        /* Configure Button EXTI line */
37        EXTI_InitStructure.EXTI_Line = BUTTON_EXTI_LINE[Button];
38        EXTI_InitStructure.EXTI_Mode = EXTI_Mode_Interrupt;
39        EXTI_InitStructure.EXTI_Trigger = EXTI_Trigger_Rising;
40        EXTI_InitStructure.EXTI_LineCmd = ENABLE;
41        EXTI_Init(&EXTI_InitStructure);
42
43        /* Enable and set Button EXTI Interrupt to the lowest priority */
44        NVIC_InitStructure.NVIC_IRQChannel = BUTTON_IRQn[Button];
45        NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 0x0F;
46        NVIC_InitStructure.NVIC_IRQChannelSubPriority = 0x0F;
47        NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;
48        NVIC_Init(&NVIC_InitStructure);
49    }
50}
51
52// Get the Pushbutton state
53uint8_t PushButtonGetState(Button_TypeDef Button) {
54    return GPIO_ReadInputDataBit(BUTTON_PORT[Button], BUTTON_PIN[Button]);
55}
56
57// This function handles EXTI0 interrupt request.
58void USER_BUTTON_EXTI_IRQHandler(void) {
59
60    //Check if EXTI_Line0 is asserted
61    if(EXTI_GetITStatus(USER_BUTTON_EXTI_LINE) != RESET) {
62        // Amb led taronja ences enviem les dades amb format per mode_prova
```



```

63     if (mode_prova == RESET){
64         mode_prova = SET;
65         LEDOn(LED_ORANGE);
66     }else{
67         mode_prova = RESET;
68         LEDOff(LED_ORANGE);
69     }
70     // We need to clear line pending bit manually
71     EXTI_ClearITPendingBit(USER_BUTTON_EXTI_LINE);
72 }
73 }

```

## fir\_fs110.h

```

1
2#include "arm_math.h"
3
4
5/* -----
6** Macro Defines
7** ----- */
8
9#define BLOCK_SIZE          550 //mostres filtrades cada cop (millor un divisor de les
    mostres totals)
10#define NUM_TAPS            198 //Nombre de coeficients del Fir (ordre+1)
11
12
13/* -----
14** Declaració de funció FIR Coeficients buffer generated using MATLAB.
15** Retornar 1 si OK
16** uint8_t fir_66Hz(uint8_t freq(de tall), uint16_t mostres, float* in(punter entrada),
    float* out(punter sortida);
17** ----- */
18uint8_t fir_fs110(float freq, uint32_t mostres, float32_t *inputF32, float32_t *outputF32);

```

## fir\_fs110.c

```

1/* -----
2** Include Files
3** ----- */
4
5#include "fir_fs110.h"
6
7/* -----
8* Declare State buffer of size ( mostres per BLOCK + nombre de coeficients - 1)
9* ----- */
10static float32_t firStateF32[BLOCK_SIZE + NUM_TAPS - 1];
11
12/* -----
13** FIR Coeficients buffer generated using MATLAB.
14** ----- */
15const float32_t firCoeffs32_2_3[NUM_TAPS] = {
16     0.00091445149005475214, -0.00016866843634103759, -0.00014337633796351326,
17     };
18
19const float32_t firCoeffs32_3_4[NUM_TAPS] = {
20     0.00062209031853593174, -0.00058277805860780312, -0.00036912877336821661,
21     };
22
23const float32_t firCoeffs32_4_5[NUM_TAPS] = {
24     -0.00080811033075463973, -0.00048552399712837289, 0.00015551639822265951,
25     };

```

\*nota: Els coeficients dels filtres s'adjunten a l'annex III.

```
25     };
26
27  /* -----
28  *  FUNCIÓ FILTRAT
29  *  ----- */
30  uint8_t fir_fs110(float freq, uint32_t mostres, float32_t *inputF32, float32_t *outputF32)
31  {
32      // Declaració de variables
33      uint32_t i;
34      uint32_t numBlocks = mostres/BLOCK_SIZE;
35      arm_fir_instance_f32 S; //Matriu d'estructura del Fir
36      uint32_t blockSize = BLOCK_SIZE;
37      //NUM_TAPS = 198
38      //BLOCK_SIZE = 550
39
40
41      // Inicialització filtre en funció dela freqüència
42
43      if (freq >= 2 && freq < 3){
44          arm_fir_init_f32(&S, NUM_TAPS, (float32_t *)&firCoeffs32_2_3[0], &firStateF32[0],
45          blockSize);
46          printf ("\rInicialitzat filtre 1");
47      }
48      if (freq >= 3 && freq <=4){
49          arm_fir_init_f32(&S, NUM_TAPS, (float32_t *)&firCoeffs32_3_4[0], &firStateF32[0],
50          blockSize);
51          printf ("\rInicialitzat filtre 2");
52      }
53      if (freq > 4 && freq <=5){
54          arm_fir_init_f32(&S, NUM_TAPS, (float32_t *)&firCoeffs32_4_5[0], &firStateF32[0],
55          blockSize);
56          printf ("\rInicialitzat filtre 3");
57      }
58      if (freq < 2 || freq > 5 ){
59          printf ("\rFrequencia fora de rang filtres (2 a 5Hz)");
60          alarma(4);
61      }else{printf ("\rFrequencia ventilador: %f Hz", freq);}
62
63      for(i=0; i < numBlocks; i++){
64          // funció de filtrat
65          arm_fir_f32(&S, &inputF32[0] + (i * blockSize), &outputF32[0] + (i * blockSize),
66          blockSize);
67      }
68      return 0;
69  }
```

## EXTI\_1\_PA1.h

```
1
2#ifndef EXTI_1_PA1_H_
3#define EXTI_1_PA1_H_
4
5
6#ifdef __cplusplus
7 extern "C" {
8#endif
9
10/* Includes -----*/
11#include "stm32f4xx.h"
12#include "../Discovery\led.h" // LED functions used in the EXTI1 interrupt handle
13#include "../SysTick\delay.h" // La usem a la interrupcio per fer proves d'activacio
14
15/* Functions -----*/
16void EXTI_1_PA1_Init(void);
17uint8_t SensorOpticGetState(void);
18
19#ifdef __cplusplus
20}
21#endif
22#endif /* exti_1_pa1_h_ */
```

## EXTI\_1\_PA1.c

```

1#include "EXTI_1_PA1.h"
2
3uint8_t pas = RESET;
4
5// Initialization of PushButton
6void EXTI_1_PA1_Init(void) {
7    GPIO_InitTypeDef GPIO_InitStructure;
8    EXTI_InitTypeDef EXTI_InitStructure;
9    NVIC_InitTypeDef NVIC_InitStructure;
10    /* Enable the BUTTON Clock */
11    RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_GPIOA, ENABLE);
12    /* Configure Button pin as input */
13    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_IN;
14    GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_NOPULL;
15    GPIO_InitStructure.GPIO_Pin = GPIO_Pin_1;
16    GPIO_Init(GPIOA, &GPIO_InitStructure);
17    /* Enable the SYSCFG Clock */
18    RCC_APB2PeriphClockCmd(RCC_APB2Periph_SYSCFG, ENABLE);
19    /* Connect EXTI Line to Button GPIO Pin */
20    SYSCFG_EXTILineConfig(EXTI_PortSourceGPIOA, EXTI_PinSource1);
21    /* Configure Button EXTI line */
22    EXTI_InitStructure.EXTI_Line = EXTI_Line1;
23    EXTI_InitStructure.EXTI_Mode = EXTI_Mode_Interrupt;
24    EXTI_InitStructure.EXTI_Trigger = EXTI_Trigger_Rising;
25    EXTI_InitStructure.EXTI_LineCmd = ENABLE;
26    EXTI_Init(&EXTI_InitStructure);
27    /* Enable and set EXTI Interrupt to the lowest priority */
28    NVIC_InitStructure.NVIC_IRQChannel = EXTI1_IRQn;
29    NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 0x0F;
30    NVIC_InitStructure.NVIC_IRQChannelSubPriority = 0x0F;
31    NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;
32    NVIC_Init(&NVIC_InitStructure);
33}
34
35// Get the SENSOR OPTIC state
36uint8_t SensorOpticGetState(void) {
37    return GPIO_ReadInputDataBit(GPIOA, GPIO_Pin_1);
38}
39
40// This function handles EXTI1 interrupt request.
41void EXTI1_IRQHandler(void) {
42    //Check if EXTI_Line0 is asserted
43    if(EXTI_GetITStatus(EXTI_Line1) != RESET) {
44
45        // Quan detectem pas d'aspa posem a 1 pas
46        // (es posarà a 0 quan l'haguem llegit)
47        pas = SET;
48        LEDToggle(LED_RED);
49        // We need to clear line pending bit manually
50
51        EXTI_ClearITPendingBit(EXTI_Line1);
52    }
53}

```

## II. Funcions creades amb Matlab.

### 1. Read\_Serial.m

% Funció per llegir les dades del port sèrie

```
function Read_Serial(mostres, frequencia) %entrem paràmetre num. mostres  
close all;
```

```
T=1/frequencia;  
t_total=T*mostres;  
t=linspace(0,t_total,mostres); %Per relacionar les mostres amb el seu temps
```

```
%Esborrar dades prèvies i tornar a declarar el port i la velocitat  
% EN CADA ORDINADOR CALDRÀ DEFINIR EL PORT QUE ASIGNI AL USB  
delete(instrfind('port'),('COM6'));  
port_serie=serial('COM6');  
port_serie.BaudRate=115200;
```

```
%obrir el port  
fopen(port_serie);  
mostra=1; %variable que ens donarà el núm. de mostra rebuda
```

```
%Al final de la lectura presentarem les dades en una gràfica  
figure('Name', 'Grafica valors');  
title('GRAFICA ACCELERACIONS');  
xlabel('temps [s]');  
ylabel('Acceleració');  
grid on;  
hold on;  
xlim([0 t_total]);
```

```
%Adquisició de mostres  
while mostra <= mostres  
    vector_accel_x(mostra)= fscanf(port_serie, '%f');  
    vector_accel_y(mostra)= fscanf(port_serie, '%f');  
    vector_accel_z(mostra)= fscanf(port_serie, '%f');  
    vector_pas(mostra)= fscanf(port_serie, '%f');  
    % Si es vol fer la gràfic a mesura que es llegeixen les dades  
    %t_plot(mostra)=t(mostra);  
    %plot(t_plot,vector_accel_x);  
    %plot(t_plot,vector_accel_y, 'g');  
    %plot(t_plot,vector_accel_z, 'r');  
    %plot(t_plot,vector_pas, 'c');  
    %drawnow;  
    mostra=mostra+1;  
end
```

```
%Plotejar  
plot(t,vector_accel_x, 'b');  
plot(t,vector_accel_y, 'g');  
plot(t,vector_accel_z, 'r');  
plot(t,vector_pas, 'k');
```

```
%tancar i esborrar el port
fclose(port_serie);
delete(port_serie);
%clear all; %Si volem esborrar la variable ans

%guardar dades en fitxer
Save_File([vector_accel_x' vector_accel_y' vector_accel_z' vector_pas'], './senyal/prova.txt');

end
```

## 2. Save\_File.m

```
% Funció per guardar les dades al disc

% entrem per paràmetre la matriu de dades i el nom de l'arxiu
function Save_File(dades, nom_arxiu)

%obrir l'arxiu (Variable file identificador fid)
fid = fopen(nom_arxiu, 'w'); % w crea un fitxer nou per a escriptura.
    % wt indica format de text del sistema operatiu.
% escriure(identificador fitxer, 'format de les dades', matriu)
fprintf(fid, '%f ', dades);

%tancar fitxer
fclose(fid);

end
```

## 3. analisi\_senyal.m

```
%Funció per analitzar els paràmetres de la senyal
%entrem per paràmetre (nom_arxiu, num. mostres, Fs, filtrat (0 --> No),...
%      ...voltes optimes a buscar per calcular mitjanes)

function [modul, angle_corregit, periode] = analisi_senyal(nom_arxiu, mostres, Fs, filtrat, voltes)
close all;
fprintf('\n-----\nAnàlisi senyal fitxer: %s\n', nom_arxiu);

Ts=1/Fs;
t_total=Ts*mostres;
t=linspace(0,t_total,mostres); %Per relacionar les msotres amb el seu temps
retard= ceil(197*1.2); % Retard del filtre. (ordre del filtre * 1.2)
```

```
%obrir l'arxiu (Variable file identificador fid)
fid = fopen(nom_arxiu, 'r'); % r crea un fitxer nou per a escriptura.
    % rt indica format de text del sistema operatiu.
variables=4; %fitxers de 4 variables (Senyal_X, Senyal_Y, Senyal_Z, Senyal_Sensor_Optic
dades=fscanf(fid,'%f',[mostres,variables]);
%tancar fitxer
fclose(fid);

% Carregeum dades del fitxer als vectors
senyal_X=dades(:, 1);
senyal_Y=dades(:, 2);
senyal_Z=dades(:, 3);
senyal_P=dades(:, 4);

%FILTRAT
if filtrat==0
    fprintf('Filtrem la senyal amb filtre_fir_4_5_o197_110fs\n');
    senyal_X=filter(fir_4_5_fs110_o197, senyal_X);
    senyal_Y=filter(fir_4_5_fs110_o197, senyal_Y);
    senyal_Z=filter(fir_4_5_fs110_o197, senyal_Z);
    senyal_P=filter(fir_4_5_fs110_o197, senyal_P);

    dades(:, 1)=senyal_X;
    dades(:, 2)=senyal_Y;
    dades(:, 3)=senyal_Z;
    dades(:, 4)=senyal_P;
end

% Acceleracions a SI Passem de +/- 2G en 16 bits a m/s^2 -- (2*9.8/2^15)
senyal_X = senyal_X*2*9.81/32768;
senyal_Y = senyal_Y*2*9.81/32768;

% Mòdul suma vectorial X-Y
senyal_XY=(senyal_X.^2+senyal_Y.^2).^0.5;

% Redimensionar amplitud vector pas per graficar
senyal_P = senyal_P * max(senyal_XY) / max(senyal_P);

% Calculo el periode del ventilador
posicio_pas_aspa = zeros(1, voltes);
%Busco les posicions del pas d'aspa
j=1;
i = retard;
while (i < mostres)
    %Busco els pasos per maxim del sensor òptic.
    if ( (senyal_P(i)-senyal_P(i-1) > 0) && (senyal_P(i+1)-senyal_P(i) <= 0) )
        posicio_pas_aspa(j) = i;
        j=j+1;
    end
    i=i+1;
end
j=j-1;
```

```

% Calculo el periode del ventilador
periode = (posicio_pas_aspa(j)- posicio_pas_aspa(1)) / ((j-1)*Fs);
fprintf('Període \t\t= %f s\n', periode);

%Grafiquem espectre potencia senyal de pas
% vector de frecuencies angulars %T=1/Fs; %df=1/(mostres/Fs);
f=(0:mostres-1)/(mostres/Fs);
ftt_accel_X=fft(senyal_X);
power_accel_X=abs(ftt_accel_X).^2;
ftt_accel_Y=fft(senyal_Y);
power_accel_Y=abs(ftt_accel_Y).^2;
ftt_accel_Z=fft(senyal_Z);
power_accel_Z=abs(ftt_accel_Z).^2;
ftt_sensor_optic=fft(senyal_P);
power_sensor_optic=abs(ftt_sensor_optic).^2;

figure('Name', 'ESPECTRE SENYAL FILTRADA');
title('Espectre de potència');
hp=plot(1:1,1:1, 'b', 1:1,2:2, 'g', 1:1, 3:3, 'k', 'DisplayName', {'Power Accel X', 'Power Accel Y', 'Power pas aspa'});
legend(hp)

hold on
plot(f,power_accel_X, 'b')
plot(f,power_accel_Y, 'g')
%plot(f,power_accel_Z, 'r')
plot(f,power_sensor_optic, 'k')
xlabel('Freqüència [Hz]')
ylabel('Amplitud')
xlim([f(1) f(ceil(length(f)/2))]);

% Grafica senyal
figure('Name', 'Grafica Acceleracions - Temps');
title('Senyal filtrada');
hp=plot(1:1,1:1, 'b', 1:1,2:2, 'g', 1:1, 3:3, 'k', 1:1, 4:4, 'c', 'DisplayName', {'Accel X', 'Accel Y', 'Accel X+Y', 'pas aspa'});
legend(hp)
coordenadesY = max(max(abs(senyal_X))*1.1, max(abs(senyal_Y))*1.1);
ylim([-coordenadesY coordenadesY]);
xlim([t(1) t(mostres)]);
xlabel('Temps [s]');
ylabel('Acceleració [m/s^2]');
grid on;
hold on;

plot(t, senyal_P, 'c');
plot(t, senyal_X, 'b');
plot(t, senyal_Y, 'g');
plot(t, senyal_XY, 'k')
%plot(t(posicio_pas_aspa), accel_XYf(posicio_pas_aspa), 'ro');

```

### % Anàlisi de senyals filtrades

#### % Calculo el nombre de mostres que analitzaré

temps\_integracio = voltes \* periode; %unitats en segons

mostres\_temps\_integracio = ceil(temps\_integracio \* Fs);

#### %Comprovo si tinc prou mostres

if (mostres\_temps\_integracio + retard > mostres)

fprintf('No hi ha prou mostres per calcular sobre aquestes voltes');

mostres\_temps\_integracio

mostres\_temps\_integracio = mostres - retard

end

### % MODUL I ANGLE BUSCANT VOLTES ÒPTIMES: MÍNIMA VARIANÇA DESFASE (Y-X)-90º

#### % Faig un vector variància de DESFASE

desfase = phase(senyal\_Y, retard) - phase(senyal\_X, retard);

#### %rectifiquem voltes perdudes i valors final per manca d'un vector

final\_vectors\_varianza=mostres;

for i=1:mostres

if desfase(i) > pi

desfase(i:mostres)=desfase(i:mostres) - (2\*pi);

end

if desfase(i) < -pi

desfase(i:mostres)=desfase(i:mostres) + (2\*pi);

end

if abs(desfase(i)) > 2\*pi

desfase(i:mostres)=0;

final\_vectors\_varianza=i; % Anotem on acaben les dades utilitzables

end

end

#### % Gràfica anàlisi senyal

fig3=figure('Name', 'Grafica anàlisi senyal');

title('Desfassament Y-X');

hp=plot(1:1:1,1:1, 'b', 1:1,2:2, 'c', 1:1, 3:3, 'ko', 1:1, 4:4, 'r', 'DisplayName', {'Desfase Y-X', 'var(desfase)', 'voltes', 'tram òptim'});

legend(hp)

xlim([t(1) t(mostres)]);

xlabel('temps [s]');

ylabel('Desfase Y-X [º]');

grid on;

hold on;

#### % Marquem les voltes segons el pas d'aspa

plot(t(posicio\_pas\_aspa), 0, 'ko');

plot(t, desfase\*180/pi, 'b');

#### % Escalem les altres senyals al desfase

escala=max(abs(desfase\*180/pi));

#### % Faig un vector per calcular la variància respecte a PI/2

i=retard;

varianza\_desfase(1:mostres)=0;

while (i + mostres\_temps\_integracio < mostres)

for j=i:mostres\_temps\_integracio



```

    varianza_desfase(i) = varianza_desfase(i) + (desfase(j) - (pi/2))^2;
end
varianza_desfase(i) = varianza_desfase(i)/mostres_temps_integracio;
i=i+1;
end
if i-1 < final_vectors_varianza
    final_vectors_varianza=i-1;
end
plot(t(retard:final_vectors_varianza),varianza_desfase(retard:final_vectors_varianza)*escala/max(abs(v
arianza_desfase)), 'c', 'LineWidth',2);

%Busco el tram on varianza_de_desfase respecte a pi/2 és mínima
[mostres_optimes, j] = min(varianza_desfase(retard : final_vectors_varianza));
posicio_optima = j + retard;

plot(t(posicio_optima:posicio_optima+mostres_temps_integracio),
desfase(posicio_optima:posicio_optima+mostres_temps_integracio)*escala/max(abs(desfase)), 'r',
'LineWidth',2);
plot(t(posicio_optima), varianza_desfase(posicio_optima)*escala/max(abs(varianza_desfase)), 'ro',
'LineWidth',2);

% Grafica senyal analitzada
fig3=figure('Name', 'Grafica senyal òptima');
title('Senyal òptima');
hp=plot(1:1,1:1, 'b', 1:1,2:2, 'g', 1:1,3:3, 'r', 1:1,4:4, 'm', 'DisplayName', {'Accel X', 'Accel Y', 'Mòdul', 'M.
comp.'});
legend(hp)
xlim([t(posicio_optima) t(posicio_optima+mostres_temps_integracio)]);
xlabel('Temps [s]');
ylabel('Acceleració [m/s^2]');
grid on;
hold on;
plot (t(posicio_optima : posicio_optima+mostres_temps_integracio), senyal_X(posicio_optima :
posicio_optima+mostres_temps_integracio), 'b')
plot (t(posicio_optima : posicio_optima+mostres_temps_integracio), senyal_Y(posicio_optima :
posicio_optima+mostres_temps_integracio), 'g')

% Assignem per reduir el nom de les variables
po=posicio_optima;
mti=po+mostres_temps_integracio;

%Calculo el MÒDUL D'AMPLITUD (integrant --> mitjana)
modul=median(senyal_XY(po:mti));

%Calculo l'ÀNGLE de l'acceleració respecte al moment de detecció del sensor òptic
j=1;
while posicio_pas_aspa(j) < po %Exploro les deteccions del pas d'aspa fins a la següent a la posició
òptima
    j=j+1;
end

angle = 0;
for i=j:voltes-1
    angle= angle + atan2(senyal_Y(posicio_pas_aspa(i)), senyal_X(posicio_pas_aspa(i)));

```

```
%fprintf('angle= %.20f Radians\n', atan2(senyal_Y(posicio_pas_aspa(i)),
senyal_X(posicio_pas_aspa(i))));
plot(t(posicio_pas_aspa(i)), senyal_XY(posicio_pas_aspa(i)), 'ro');
end
angle=angle/voltes;
fprintf('Mòdul \t\t\t= %fm/s^2\t\tangle \t\t\t= %.2fº\n', modul, angle*180/pi);

% PER CORREGIR L'ANGLE, ABANS DE ATAN2, ESCALAR LES SENYALS A LA MITJA DELS
% DOS MAXIMS QUE ENVOLTEN EL PAS D'ASPA
angle_corregit = 0;
mitjana_maxims=0;
for i=j:voltes-1
    %faig la mitjana de dos màxims al voltant del pas d'aspa (dos cicles)
    Y_max = (max(senyal_Y(posicio_pas_aspa(i)-
floor(periode*Fs):posicio_pas_aspa(i)+floor(periode*Fs)))+ max(senyal_Y(posicio_pas_aspa(i+1)-
floor(periode*Fs):posicio_pas_aspa(i+1)+floor(periode*Fs)))/2;
    X_max = (max(senyal_X(posicio_pas_aspa(i)-
floor(periode*Fs):posicio_pas_aspa(i)+floor(periode*Fs)))+ max(senyal_X(posicio_pas_aspa(i+1)-
floor(periode*Fs):posicio_pas_aspa(i+1)+floor(periode*Fs)))/2;
    maxim=max(Y_max, X_max);
    mitjana_maxims=mitjana_maxims+maxim;
    % Calculo l'angle escalant les senyals X o Y al valor màxim
    angle_corregit= angle_corregit + atan2(senyal_Y(posicio_pas_aspa(i)) * maxim/Y_max,
senyal_X(posicio_pas_aspa(i)) * maxim/X_max);
    %fprintf('angle_afinat= %.20f Radians\n', atan2(senyal_Y(posicio_pas_aspa(i)) * X_max/Y_max,
senyal_X(posicio_pas_aspa(i))));
    plot(t(posicio_pas_aspa(i)), senyal_XY(posicio_pas_aspa(i)), 'ro');
end
angle_corregit=angle_corregit/voltes;
mitjana_maxims=mitjana_maxims/voltes;

% Per corregir el mòdul, l'escalo a la mitjana
modul_corregit=median( ((senyal_X(po:mti)*mitjana_maxims/max(senyal_X(po:mti))).^2 +
(senyal_Y(po:mti)*mitjana_maxims/max(senyal_Y(po:mti))).^2).^0.5 );

plot(t(po:mti), modul, 'r');
%plot(t(po:mti), angle*180/pi, '--m');
plot([t(posicio_pas_aspa(i)) t(posicio_pas_aspa(i))], [-modul modul], 'r');
plot (t(po:mti), senyal_X(po:mti).* mitjana_maxims/max(senyal_X(po:mti)), 'b--')
plot (t(po:mti), senyal_Y(po:mti).* mitjana_maxims/max(senyal_Y(po:mti)), 'g--')
plot(t(po:mti), modul_corregit, 'm');

fprintf('Mòdul corregit\t= %fm/s^2\t\tangle corregit\t= %.2fº\n', modul_corregit,
angle_corregit*180/pi);

% DESCOMENTAR SI VULL VISUALITZAR LES ACCELERACIONS CADA N VOLTES
%evolucio_temps(dades, mostres, Fs, varianza_desfase, posicio_optima, mostres_temps_integracio)

fprintf('-----\n');

end
```

## 4. fir\_2\_3\_fs110\_o197.m

```
function Hd = fir_2_3_fs110_o197
%FIR_2_3_FS110_O197 Returns a discrete-time filter object.

%
% MATLAB Code
% Generated by MATLAB(R) 7.12 and the DSP System Toolbox 8.0.
%
% Generated on: 07-Jun-2015 10:50:09

% Equiripple Bandpass filter designed using the FIRPM function.

% All frequency values are in Hz.
Fs = 110; % Sampling Frequency

Fstop1 = 1; % First Stopband Frequency
Fpass1 = 2; % First Passband Frequency
Fpass2 = 3; % Second Passband Frequency
Fstop2 = 4; % Second Stopband Frequency
Dstop1 = 0.001; % First Stopband Attenuation
Dpass = 0.1; % Passband Ripple
Dstop2 = 0.001; % Second Stopband Attenuation
dens = 20; % Density Factor

% Calculate the order from the parameters using FIRPMORD.
[N, Fo, Ao, W] = firpmord([Fstop1 Fpass1 Fpass2 Fstop2]/(Fs/2), [0 1 ...
    0], [Dstop1 Dpass Dstop2]);

% Calculate the coefficients using the FIRPM function.
b = firpm(N, Fo, Ao, W, {dens});
Hd = dfilt.dfir(b);

% [EOF]
```

## 5. fir\_3\_4\_fs110\_o197.m

```
function Hd = fir_3_4_fs110_o197
%FIR_3_4_FS110_O197 Returns a discrete-time filter object.

%
% MATLAB Code
% Generated by MATLAB(R) 7.12 and the DSP System Toolbox 8.0.
%
% Generated on: 07-Jun-2015 10:46:55

% Equiripple Bandpass filter designed using the FIRPM function.

% All frequency values are in Hz.
Fs = 110; % Sampling Frequency

Fstop1 = 2; % First Stopband Frequency
```

```
Fpass1 = 3; % First Passband Frequency
Fpass2 = 4; % Second Passband Frequency
Fstop2 = 5; % Second Stopband Frequency
Dstop1 = 0.001; % First Stopband Attenuation
Dpass = 0.1; % Passband Ripple
Dstop2 = 0.001; % Second Stopband Attenuation
dens = 20; % Density Factor

% Calculate the order from the parameters using FIRPMORD.
[N, Fo, Ao, W] = firpmord([Fstop1 Fpass1 Fpass2 Fstop2]/(Fs/2), [0 1 ...
    0], [Dstop1 Dpass Dstop2]);

% Calculate the coefficients using the FIRPM function.
b = firpm(N, Fo, Ao, W, {dens});
Hd = dfilt.dfir(b);

% [EOF]
```

## 6. fir\_4\_5\_fs110\_o197.m

```
function Hd = fir_4_5_fs110_o197
%FIR_4_5_FS110_O197 Returns a discrete-time filter object.

%
% MATLAB Code
% Generated by MATLAB(R) 7.12 and the DSP System Toolbox 8.0.
%
% Generated on: 07-Jun-2015 09:59:03

% Equiripple Bandpass filter designed using the FIRPM function.

% All frequency values are in Hz.
Fs = 110; % Sampling Frequency

Fstop1 = 3; % First Stopband Frequency
Fpass1 = 4; % First Passband Frequency
Fpass2 = 5; % Second Passband Frequency
Fstop2 = 6; % Second Stopband Frequency
Dstop1 = 0.001; % First Stopband Attenuation
Dpass = 0.1; % Passband Ripple
Dstop2 = 0.001; % Second Stopband Attenuation
dens = 20; % Density Factor

% Calculate the order from the parameters using FIRPMORD.
[N, Fo, Ao, W] = firpmord([Fstop1 Fpass1 Fpass2 Fstop2]/(Fs/2), [0 1 ...
    0], [Dstop1 Dpass Dstop2]);

% Calculate the coefficients using the FIRPM function.
b = firpm(N, Fo, Ao, W, {dens});
Hd = dfilt.dfir(b);

% [EOF]
```

## 7. phase.m

% Retorna la fase en radians d'una senyal sinusoidal

```
function [ phase_accel ] = phase(accel, retard )
```

```
mostres=length(accel);
phase_accel(1:mostres) = 0;
```

%Busco les posicions dels maxims

```
max_accel=1;
i = retard;
while (i < mostres)
    %Busco el pas per maxim
    if ( (accel(i)-accel(i-1) > 0) && (accel(i+1)-accel(i) <= 0) )
        posicio_max_accel(max_accel) = i;
        max_accel=max_accel+1;
    end
    i=i+1;
end
max_accel=max_accel-1;
```

%Faig vector phase

```
j=1;
while ( j < max_accel);
    phase_accel(posicio_max_accel(j) : posicio_max_accel(j+1)) = linspace( 2*pi*(j-1), 2*pi*j,
    posicio_max_accel(j+1) - posicio_max_accel(j) +1);
    j=j+1;
end
```

% Descomentar per presentar en gràfica

```
% t=linspace(0,10,mostres);
% figure(3)
% hold on
% plot (t,phase_accel)
```

```
end
```

## 8. evolucion\_temps.m

% Funció per visualitzar l'evolució de les acceleracions, velocitat...

```
function evolucion_temps(dades, mostres, Fs, optim, posicio_optima, mostres_integracio_voltes)
%close all;
Ts=1/Fs;
t=linspace(0,Ts*mostres,mostres); %Creem vector de temps per graficar
%vector de freqüències angulars
f=(0:mostres-1)/(mostres*Ts); %df=1/(mostres*Ts);
retard=ceil(197*1.2); % Part de senyal afectada pel retard del Fir ordre 197
```

% Separem les dades de la matriu en vectors

```
senyal_X=dades(:, 1);  
senyal_Y=dades(:, 2);  
senyal_Z=dades(:, 3);  
senyal_P=dades(:, 4);
```

% Acceleracions a SI Passem de +/- 2G en 16 bits a m/s^2 -- (2\*9.8/2^15)

```
senyal_X = senyal_X*2*9.81/32768;  
senyal_Y = senyal_Y*2*9.81/32768;
```

% Mòdul suma vectorial X-Y

```
senyal_XY=(senyal_X.^2+senyal_Y.^2).^0.5;
```

% Redimensionar vector pas per graficar

```
senyal_P = senyal_P * max(senyal_XY) / max(senyal_P);
```

% Graficar: configuració del gràfic

```
fig1=figure('Name', 'Grafica acceleracions');  
set (fig1, 'Units', 'normalized', 'Position', [0.49,0,0.49,1]);
```

```
xlim([t(1) t(mostres)]);  
y1 = -max(senyal_XY)*.9; y2 = max(senyal_XY)*1.1;  
ylim([y1, y2]);
```

%creem una linia a la gràfica d'acceleracions

```
figure(fig1);  
hl=findobj('tag', 'linia');  
if isempty(hl)  
    hl = plot([0 0], [y1 y2], 'tag', 'linia');  
end
```

```
grid on;  
hold on;  
title('GRAFICA ACCELERACIONS-TEMPS');  
xlabel('temps [s]');  
ylabel('Acceleració [m/s^2]');
```

```
plot(t, senyal_X, 'b');  
plot(t, senyal_Y, 'g');  
plot(t, senyal_P, 'c');  
plot(t, senyal_XY, 'k');
```

%grafiquem optim escalat a y2= max(senyal\_XY)\*1.1;

```
plot(t, optim*y2/max(optim), 'r', 'LineWidth', 2);  
plot(t(posicio_optima), optim(posicio_optima)*y2/max(optim), 'ko', 'LineWidth', 2);
```

% VELOCITATS (INTEGREM LES ACCELERACIONS)

```
senyal_XV = senyal_X; %per dimensionar l'array  
senyal_YV = senyal_Y; %per dimensionar l'array  
for i=1:mostres-1  
    senyal_XV(i+1) = senyal_XV(i) + senyal_X(i)*1/Fs + 3.8*10^-7;  
    senyal_YV(i+1) = senyal_YV(i) + senyal_Y(i)*1/Fs - 4.5*10^-6;  
end
```

```

% POSICIONS (INTEGREM LES VELOCITATS)
senyal_XP = senyal_X; %per dimensionar l'array
senyal_YP = senyal_Y; %per dimensionar l'array
for i=1:mostres-1
    senyal_XP(i+1) = senyal_XP(i) + senyal_XV(i)*1/Fs;
    senyal_YP(i+1) = senyal_YP(i) + senyal_YV(i)*1/Fs;
end

% VARIABLES PER RECOLOCAR VELOCITAT I POSICIO AL CENTRE DE LA GRÀFICA
correccio_XV = 0; correccio_YV = 0; correccio_XP = 0; correccio_YP = 0;

% GRAFICA per acceleracions, velocitat i posició (polar)
fig2=figure('Name', 'Gràfica vectors (polar)');
set (fig2, 'Units', 'normalized', 'Position', [0,0,0.5,1]);
coordenades = max(max(senyal_X), max(senyal_Y));

i=posicio_optima;
while (i > retard + mostres_integracio_voltes)
    i = i - mostres_integracio_voltes;
end
retard=i; k=0;
for i=retard:mostres-1
    figure(fig2);
    hold on;
    title('ACCELERACIÓ XY');
    xlabel('Temps [s]');
    ylabel('Acceleració [m/s^2]');
    xlim([-coordenades coordenades]);
    ylim([-coordenades coordenades]);
    k=k+1;
    if mod(k, mostres_integracio_voltes) == 0
        hold off
        correccio_XV = senyal_XV(i);
        correccio_YV = senyal_YV(i);
        correccio_XP = senyal_XP(i);
        correccio_YP = senyal_YP(i);
        % Si volem veure només un segon de senyal a fig1, descomentar
        % figure(fig1);
        % xlim([t(i) t(i+Fs)]);
        % figure(fig2);
        pause(2)
    end

    % plot(senyal_X(i),senyal_Y(i), 'ro');
    plot((senyal_XV(i)-correccio_XV)*20, (senyal_YV(i)-correccio_YV)* 20, 'bo');
    plot((senyal_XP(i)-correccio_XP)*200, (senyal_YP(i)-correccio_YP)*200, 'kx');
    set (hl, 'XData', [t(i) t(i)], 'YData', [y1 y2], 'Color', 'r');

    pause(.02)

end

end

```

## 9. calcul\_contrapesos.m

% Funció per calcular els contrapesos a partir de vibració inicial i  
% vibració amb contrapés

**function** calcul\_contrapesos(arxiu\_Ao, arxiu\_Ap, mostres, Fs, filtrat, voltes)  
close all;

massa = [0.00230, 0.00306, 0.00392, 0.00410, 0.00574, 0.0078, 0.0085, 1];  
massa\_pes\_prova=[0.00392];  
RADI\_PES\_PROVA = 0.5;  
RADI\_ASPIA = 0.5;

%: Sumar un gram a tot per la cinta  
% /\* Seqüència: posició vector, pes [kg], monedes:  
% 1 0.00230 --> 1 x 1 cèntim  
% 2 0.00306 --> 1 x 2 cèntims  
% 3 0.00392 --> 1 x 5 cèntims  
% 4 0.00410 --> 1 x 10 cèntim  
% 5 0.00574 --> 1 x 20 cèntims  
% 6 0.00780 --> 1 x 50 cèntims  
% 7 0.00850 --> 1 x 2 euros  
% 8 --> ERROR!  
%  
% PES DE PROVA  
% 1 0.00392 --> 1 x 5 cèntims  
% \*/

[modul\_inicial, angle\_inicial, periode\_inicial] = analisi\_senyal(arxiu\_Ao, mostres, Fs, filtrat, voltes);

[modul\_prova, angle\_prova, periode\_prova] = analisi\_senyal(arxiu\_Ap, mostres, Fs, filtrat, voltes);

fprintf('\n----- Càlcul contrapesos ----- \n');  
fprintf('Mòdul inicial:\t%f m/s^2\t Angle inicial:\t%+3.1fº\tPeriode inicial: %.4f s\n', modul\_inicial,  
angle\_inicial\*180/pi, periode\_inicial);  
fprintf('Mòdul pes prova:%f m/s^2\t Angle contrap:\t%+3.1fº\tPeriode contrap: %.4f s\n', modul\_prova,  
angle\_prova\*180/pi, periode\_prova);

%Càlcul del pes estimat en base al canvi que fa el pes de prova

diferencia\_periodes = periode\_prova - periode\_inicial;

wo = 2\*pi/periode\_inicial;

wp = 2\*pi/periode\_prova;

%DETERMINAR LA MASSA DELS CONTRAPESOS

% Quan ha augmentat l'acceleració en l'eix Y?

% No hauria d'haver augmentat. Indica un desfassament del sensor òptic

% Si AppY > 0 vol dir que quan "veig" l'aspa, ja ha passat el pes prova

% Si AppY < 0 vol dir que el sensor detecta abans que el pes arribi



```

AppYo = (modul_prova * sin(angle_prova)) - (modul_inicial * sin(angle_inicial));
% Quan ha augmentat l'acceleració en l'eix X?
AppXo = (modul_prova * cos(angle_prova)) - (modul_inicial * cos(angle_inicial));
fprintf('Comprovació variació accel. Y per contrapes: %f\tVariació accel. X: %f\n', AppYo, AppXo);
fprintf('Si variació accel. Y >> 0 ==> avançar sensor òptic. En cas contrari retrassar.\n ');

% Calculem desviació sensor òptic respecte a l'eix X
App=(AppXo^2+AppYo^2)^.5; %Modul acceleració contrapés
desviacio_sensor_optic=asin(AppYo/App);
% Graus per mostra
graus_per_mostra=360/(periode_inicial*Fs);

fprintf('desviacio_sensor_optic: %.2fº \tGraus per mostra: %.2fº\n', desviacio_sensor_optic*180/pi,
graus_per_mostra);

%Corregim valor AppX per desviació sensor òptic
AppX=App; % Tota la variació del mòdul pel contrapés s'aplica a l'eix X
%Si volem corregir l'angle podem usar l'expressió següent, però pot introduir error.
%Millor corregir físicament l'alineació del sensor òptic amb l'eix X
%angle_inicial=angle_inicial-desviacio_sensor_optic; %Corregim l'angle mesurat sense pes de prova.

% Força que genera la massa del pes de prova, per una distància R, a una
% velocitat angular w. Força pes de prova Fpp = mc·An = (mc·R·w^2)
Fpp = massa_pes_prova(1) * RADI_PES_PROVA * wp^2;

% Massa equivalent = Força pes de prova/Acceleració pes de prova
Me = Fpp / AppX;

% La força que genera el desequilibri (la que cal corregir amb contrapesos)
% és la força que aplicada a la massa equivalent contrarrestarà les acceleracions
% originals.
Fc = Me * modul_inicial;

fprintf('\nForça pes de prova [Fpp = mpp·(R·w^2)]: %f N \tMassa equivalent: %.2f kg\nForça per
contrarrestar: Fc = Ao·(Fpp/varAx pel pes prova): %f N\n', Fpp, Me, Fc);

%Comprovem que el mòdul hagi variat entre el 30% i el 100% pel pes de prova
if ((modul_prova-modul_inicial) / modul_inicial < 0.3 || (modul_prova-modul_inicial) / modul_inicial >
1)
    fprintf('\nVARIACIÓ DEL MÒDUL INCORRECTA: %.2f%% (ha de ser entre 30%% i 100%%)\n',
(modul_prova-modul_inicial)*100 / modul_inicial)
else
    fprintf('\nVARIACIÓ DEL MÒDUL CORRECTA: %.2f%% (entre 30%% i 100%%)\n', (modul_prova-
modul_inicial)*100 / modul_inicial);
end

% Fc = massa(pesos) * RADI(pesos) * wo^2 * k (Afegim la constant k per si
% cal fer ajustos de linealitat (friccions rigideses...)
k=1.08; %valor obtingut de forma experimental. Aporta el millor equilibrat
fprintf('Factor correcció: k = %.2f\n', k);
% Supposem radi màxim 0.5m

```

% Massa: Augmentem el valor de la massa del contrapès fins que el puguem  
% posar dins les aspes (de 0,5m) i aconseguir la força desitjada o més

```
i=1;  
massa_X=massa(i);  
while ( massa_X < Fc * abs(cos(angle_inicial)) / (RADI_ASPA * wo^2 * k) )  
    i=i+1;  
    massa_X=massa(i);  
    % Si no dispo de prous pesos per l'acceleració detectada, indico error  
    if (i > 7) %(Si i se surt del vector)  
        fprintf('no dispo de prous pesos. ERROR\n\n');  
    end  
end
```

% Massa: Augmentem el valor del pes fins que el puguem posar dins les pales i aconseguir la força  
desitjada

```
i=1;  
massa_Y=massa(i);  
while ( massa_Y < Fc * abs(sin(angle_inicial)) / (RADI_ASPA * wo^2 * k) )  
    i=i+1;  
    massa_Y=massa(i);  
    % Si no dispo de prous pesos per l'acceleració detectada, indico error  
    if (i > 7) %(Si i se surt del vector)  
        fprintf('no dispo de prous pesos. ERROR\n\n');  
    end  
end
```

% RADI(pesos) =  $F_c / (k * massa(pesos) * wo^2)$

```
radi_pes_X = 100 * cos(pi+angle_inicial) * Fc / (massa_X * wo^2 * k);  
radi_pes_Y = 100 * sin(pi+angle_inicial) * Fc / (massa_Y * wo^2 * k);
```

```
fprintf('\n* * * SORTIDA * * *\n');  
fprintf('Contrapès X: %.2f g \tRadi X: %.2f cm\nContrapès Y: %.2f g \tRadi Y: %.2f cm\n', massa_X*1000,  
radi_pes_X, massa_Y*1000, radi_pes_Y);  
fprintf('* * * * FINAL * * *\n');
```

```
end
```

### III. Coeficients dels filtres FIR

A la figura 41 es mostra l'entorn de disseny de filtres de l'eina FDATool de Matlab, des de la qual es poden exportar els coeficients del filtre. Els 3 filtres utilitzats s'han creat amb els mateixos paràmetres, canviant únicament en 1 Hz les freqüències de pas i de tall.

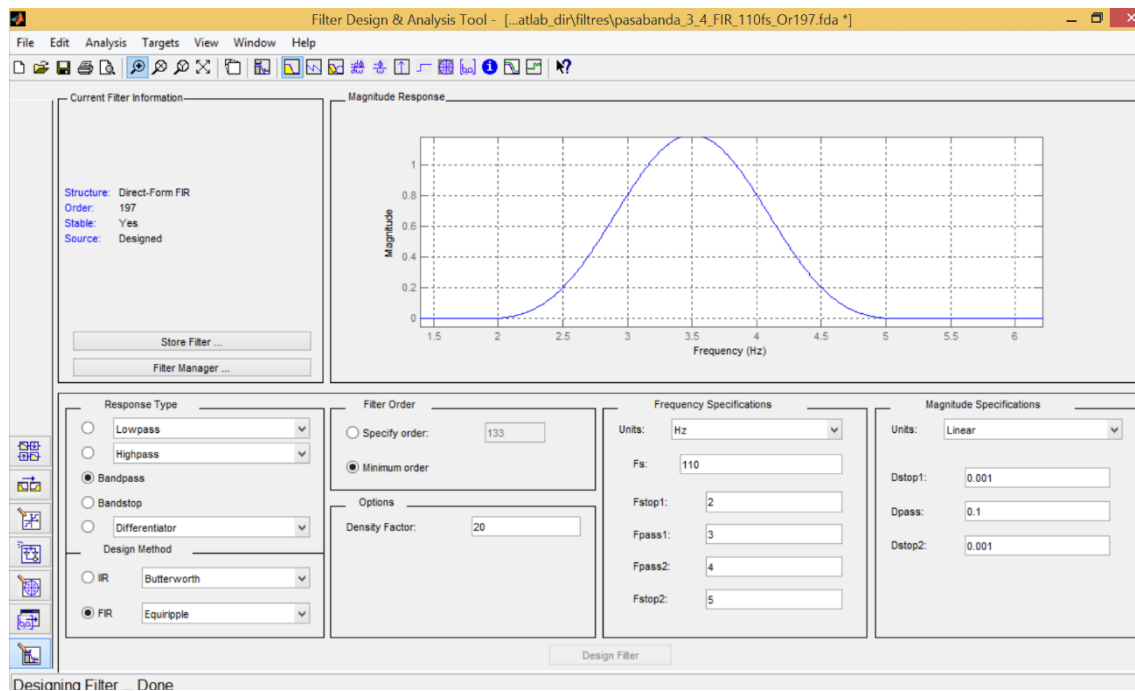


Figura 40: Disseny de filtre FIR passa banda 3-4Hz amb FDATool

## Filtre FIR passabanda de 2 a 3 Hz per freqüència de mostreig de 110Hz i ordre 197

0.00091445149005475214	-0.0013014203747546299	0.0013088427985989508
-0.00016866843634103759	-0.004034985640285327	0.0037410307214729805
-0.00014337633796351326	-0.0068319121719140445	0.0059483033763149942
-0.00011372803521829595	-0.0096267280132457771	0.0078914978826876633
-0.000079825976863329975	-0.012351327530813682	0.009540461008209376
-0.000041658366672584846	-0.014936270814770638	0.01087412746159612
0.00000086632515465859782	-0.017312406673048942	0.011880611866023879
0.000047731514250688441	-0.019412965219676087	0.012557075672999313
0.000098616807699835663	-0.02117558682686593	0.012909236960060178
0.00015279086756278038	-0.022544115619443614	0.012950699393999853
0.00020901111478609886	-0.023470263680258029	0.01270218255672502
0.00026544136097087715	-0.023915209128079394	0.012190580298760266
0.00031960807354433466	-0.023851017502159587	0.011447807068007607
0.00036839867125772977	-0.02326175856615089	0.01050951357334793
0.00040811583232241381	-0.022144257944752232	0.0094138109553199949
0.00043457963602755987	-0.02050849077873378	0.0082001187620067088
0.0004432569480485721	-0.018377684984481016	0.006908109165687194
0.00042944191798923901	-0.015788138292778225	0.0055765621046709431
0.00038849060490291	-0.01278860087142304	0.0042421060242833998
0.00031601644005172728	-0.0094391310546522618	0.0029381892897477734
0.00020812297302529612	-0.0058097441353938319	0.0016948258511317381
0.000062136171745307391	-0.0019790779046787057	0.00053858479753248769
-0.00012516072259824288	0.0019681935650466146	-0.00051031805804045423
-0.00035133712972890455	0.005942437136227407	-0.0014347033349575
-0.00061925296516473008	0.0098525553755554659	-0.0022235426187469849
-0.00092390739128464191	0.013607186453379595	-0.0028699994968180795
-0.0012602912076927414	0.017117970801748417	-0.0033721091678667273
-0.0016222261240267147	0.02030152041562713	-0.0037321346715632519
-0.002001313889070367	0.023081545806319503	-0.0039561577466279531
-0.0023867786333237116	0.025391057750780273	-0.0040535997227748467
-0.0027658507140084247	0.02717420230211328	-0.0040365846321998976
-0.003124286341176757	0.028387691327602806	-0.0039192787275578646
-0.0034467692999312835	0.029001945087417727	-0.0037172413919796354
-0.0037172413919796354	0.029001945087417727	-0.0034467692999312835
-0.0039192787275578646	0.028387691327602806	-0.003124286341176757
-0.0040365846321998976	0.02717420230211328	-0.0027658507140084247
-0.0040535997227748467	0.025391057750780273	-0.0023867786333237116
-0.0039561577466279531	0.023081545806319503	-0.002001313889070367
-0.0037321346715632519	0.02030152041562713	-0.0016222261240267147
-0.0033721091678667273	0.017117970801748417	-0.0012602912076927414
-0.0028699994968180795	0.013607186453379595	-0.00092390739128464191
-0.002235426187469849	0.0098525553755554659	-0.00061925296516473008
-0.0014347033349575	0.005942437136227407	-0.00035133712972890455
-0.00051031805804045423	0.0019681935650466146	-0.00012516072259824288
0.00053858479753248769	-0.0019790779046787057	0.000062136171745307391
0.0016948258511317381	-0.0058097441353938319	0.00020812297302529612
0.0029381892897477734	-0.0094391310546522618	0.00031601644005172728
0.0042421060242833998	-0.01278860087142304	0.00038849060490291
0.0055765621046709431	-0.015788138292778225	0.00042944191798923901
0.006908109165687194	-0.018377684984481016	0.0004432569480485721
0.0082001187620067088	-0.02050849077873378	0.00043457963602755987
0.0094138109553199949	-0.022144257944752232	0.00040811583232241381
0.01050951357334793	-0.02326175856615089	0.00036839867125772977
0.011447807068007607	-0.023851017502159587	0.00031960807354433466
0.012190580298760266	-0.023915209128079394	0.00026544136097087715
0.01270218255672502	-0.023470263680258029	0.00020901111478609886
0.012950699393999853	-0.022544115619443614	0.00015279086756278038
0.012909236960060178	-0.02117558682686593	0.000098616807699835663
0.012557075672999313	-0.019412965219676087	0.000047731514250688441
0.011880611866023879	-0.017312406673048942	0.00000086632515465859782
0.01087412746159612	-0.014936270814770638	-0.000041658366672584846
0.009540461008209376	-0.012351327530813682	-0.000079825976863329975
0.0078914978826876633	-0.0096267280132457771	-0.00011372803521829595
0.0059483033763149942	-0.0068319121719140445	-0.00014337633796351326
0.0037410307214729805	-0.004034985640285327	-0.00016866843634103759
0.0013088427985989508	-0.0013014203747546299	0.00091445149005475214

## Filtre FIR passabanda de 3 a 4 Hz per freqüència de mostreig de 110Hz i ordre 197

0.00062209031853593174	0.018122242820920641	0.016485623600335959
-0.00058277805860780312	0.019078900486481754	0.01427636709637428
-0.00036912877336821661	0.019274452185530547	0.011621328308529584
-0.00023059946365991694	0.018658452802231856	0.0086594155045013402
-0.00013787175055117202	0.01721548828297214	0.005535530696173593
-0.000072710119219042218	0.014967834172044232	0.002393183085005475
-0.000024491815784599778	0.011975825871380848	-0.00063280684673607652
0.000012044757636174726	0.0083366627026652766	-0.0034201724738884371
0.0000383872727081374	0.004181415922495638	-0.0058665646131224247
0.000053393124305576786	-0.00032950643744048512	-0.0078907053280889428
0.000054490602485629666	-0.0050128964506176345	-0.0094366296151010293
0.000038234488254037949	-0.0096709063159487445	-0.010473631687886464
0.0000012413097351045592	-0.014099399638315798	-0.010996426397449247
-0.000059132443862762994	-0.018097152772983759	-0.011023191776272654
-0.00014422964942954839	-0.021475892512542526	-0.01059323517185359
-0.00025345891358514679	-0.024069142534203485	-0.0097633899086760571
-0.00038397430928178381	-0.025741126286432856	-0.0086044129717469039
-0.0005303641126562582	-0.026393185766647051	-0.0071961778844418043
-0.00068453266503061821	-0.025969878260584115	-0.0056233986150112389
-0.00083600590079424043	-0.024462506444707988	-0.0039707978269861843
-0.00097214859075981253	-0.021910832559197249	-0.0023189476565990373
-0.0010786714200015995	-0.018401810327211568	-0.00074103350518891359
-0.0011406041743147212	-0.014067026180297516	0.00069991439072302365
-0.0011430629936515617	-0.0090767956605227081	0.001953164839972106
-0.0010725989122197378	-0.0036340156073940699	0.0029820671653122974
-0.00091806336811384691	0.002035871824417162	0.0037648404227947183
-0.00067198815571281904	0.0076939497881448986	0.0042935312582295382
-0.00033148477283357811	0.013099024359920525	0.0045726574499010651
0.00010022973470317901	0.018018756647020857	0.0046183708041201777
0.00061502410669841718	0.022240006202253557	0.0044570682757974324
0.0011952561302935602	0.02557899910614124	0.0041224877812748124
0.0018213367305261326	0.027889800578030279	0.0036517639276537565
0.0024617507366737541	0.029071045441022261	0.0030844201582662157
0.0030844201582662157	0.029071045441022261	0.0024617507366737541
0.0036517639276537565	0.027889800578030279	0.0018213367305261326
0.0041224877812748124	0.02557899910614124	0.0011952561302935602
0.0044570682757974324	0.022240006202253557	0.00061502410669841718
0.0046183708041201777	0.018018756647020857	0.00010022973470317901
0.0045726574499010651	0.013099024359920525	-0.00033148477283357811
0.0042935312582295382	0.0076939497881448986	-0.00067198815571281904
0.0037648404227947183	0.002035871824417162	-0.00091806336811384691
0.0029820671653122974	-0.0036340156073940699	-0.0010725989122197378
0.001953164839972106	-0.0090767956605227081	-0.0011430629936515617
0.00069991439072302365	-0.014067026180297516	-0.0011406041743147212
-0.00074103350518891359	-0.018401810327211568	-0.0010786714200015995
-0.0023189476565990373	-0.021910832559197249	-0.00097214859075981253
-0.0039707978269861843	-0.024462506444707988	-0.00083600590079424043
-0.0056233986150112389	-0.025969878260584115	-0.00068453266503061821
-0.0071961778844418043	-0.026393185766647051	-0.0005303641126562582
-0.0086044129717469039	-0.025741126286432856	-0.00038397430928178381
-0.0097633899086760571	-0.024069142534203485	-0.00025345891358514679
-0.01059323517185359	-0.021475892512542526	-0.00014422964942954839
-0.011023191776272654	-0.018097152772983759	-0.000059132443862762994
-0.010996426397449247	-0.014099399638315798	0.0000012413097351045592
-0.010473631687886464	-0.0096709063159487445	0.000038234488254037949
-0.0094366296151010293	-0.0050128964506176345	0.000054490602485629666
-0.0078907053280889428	-0.00032950643744048512	0.000053393124305576786
-0.0058665646131224247	0.004181415922495638	0.0000383872727081374
-0.0034201724738884371	0.0083366627026652766	0.000012044757636174726
-0.00063280684673607652	0.011975825871380848	-0.000024491815784599778
0.002393183085005475	0.014967834172044232	-0.000072710119219042218
0.005535530696173593	0.01721548828297214	-0.00013787175055117202
0.0086594155045013402	0.018658452802231856	-0.00023059946365991694
0.011621328308529584	0.019274452185530547	-0.00036912877336821661
0.01427636709637428	0.019078900486481754	-0.00058277805860780312
0.016485623600335959	0.018122242820920641	0.00062209031853593174

## Filtre FIR passabanda de 4 a 5 Hz per freqüència de mostreig de 110Hz i ordre 197

-0.00080811033075463973	-0.008880435133578099	-0.012350028710068326
-0.00048552399712837289	-0.0045833416195247894	-0.014804899842927034
0.00015551639822265951	0.00028164113707657363	-0.016138999035742615
-0.00032614311638072611	0.0053998161373836669	-0.016329268067457929
0.000097931011619376679	0.010420578275316256	-0.015435600216143475
-0.00018495293315187017	0.014982198281539464	-0.013587755738108276
0.000092519431352912644	0.018738870200625313	-0.010975204385964565
-0.000083295058831828672	0.021386066927599597	-0.0078272897405748942
0.000079922372676111825	0.022686637509183528	-0.0043959778562702749
-0.000053255149295052006	0.022489153991485138	-0.00093502020893060475
0.000015539763380733262	0.020745037596271047	0.0023178947016936854
-0.00010816986453272273	0.017514419075915496	0.0051576796953758291
-0.00010628611738582982	0.012966577561281858	0.0074226692837178407
-0.00021886462955307269	0.0073721837693906484	0.009006277500215442
-0.00023854270871311826	0.0010850476577876805	0.0098568993931012101
-0.00030765778865298138	-0.005478663610001361	0.0099797550053203204
-0.00028975643321109466	-0.011871141511785054	0.0094314008289602923
-0.00026925402602539247	-0.017642616370917023	0.0083100813885446205
-0.00015971662398373086	-0.022376233727909957	0.0067476670327635642
-0.000016850733695144681	-0.025716993177712415	0.0048939830342465407
0.00020608128045978245	-0.027400983476996002	0.002906033752327519
0.0004628711494843264	-0.027275085570301824	0.00093561621911550526
0.00076746243632493159	-0.025311491632806951	-0.00088228208007249537
0.0010727571544489603	-0.021611598991037977	-0.0024374874570747183
0.0013668163142971654	-0.01640078050958085	-0.0036493754250992855
0.001600147947849902	-0.010014602161730363	-0.0044728623973699531
0.0017481772799190533	-0.0028764413779525655	-0.004894573414889334
0.0017684491898499288	0.0045332946690095259	-0.0049323425849908745
0.0016394218391721461	0.011708683446092147	-0.0046316948764253425
0.0013356902430489617	0.01815685768836435	-0.0040557190276451181
0.00085612679036957927	0.023431606988525026	-0.003282068778412273
0.00020743205054079596	0.027166552870375647	-0.0023928417996881891
-0.00058141661334425816	0.029101516473423631	-0.0014690303235575458
-0.0014690303235575458	0.029101516473423631	-0.00058141661334425816
-0.0023928417996881891	0.027166552870375647	0.00020743205054079596
-0.003282068778412273	0.023431606988525026	0.00085612679036957927
-0.0040557190276451181	0.01815685768836435	0.0013356902430489617
-0.0046316948764253425	0.011708683446092147	0.0016394218391721461
-0.0049323425849908745	0.0045332946690095259	0.0017684491898499288
-0.004894573414889334	-0.0028764413779525655	0.0017481772799190533
-0.0044728623973699531	-0.010014602161730363	0.001600147947849902
-0.0036493754250992855	-0.01640078050958085	0.0013668163142971654
-0.0024374874570747183	-0.021611598991037977	0.0010727571544489603
-0.00088228208007249537	-0.025311491632806951	0.00076746243632493159
0.00093561621911550526	-0.027275085570301824	0.0004628711494843264
0.002906033752327519	-0.027400983476996002	0.00020608128045978245
0.0048939830342465407	-0.025716993177712415	-0.000016850733695144681
0.0067476670327635642	-0.022376233727909957	-0.00015971662398373086
0.0083100813885446205	-0.017642616370917023	-0.00026925402602539247
0.0094314008289602923	-0.011871141511785054	-0.00028975643321109466
0.0099797550053203204	-0.005478663610001361	-0.00030765778865298138
0.0098568993931012101	0.0010850476577876805	-0.00023854270871311826
0.009006277500215442	0.0073721837693906484	-0.00021886462955307269
0.0074226692837178407	0.012966577561281858	-0.00010628611738582982
0.0051576796953758291	0.017514419075915496	-0.00010816986453272273
0.0023178947016936854	0.020745037596271047	0.000015539763380733262
-0.00093502020893060475	0.022489153991485138	-0.000053255149295052006
-0.0043959778562702749	0.022686637509183528	0.000079922372676111825
-0.0078272897405748942	0.021386066927599597	-0.000083295058831828672
-0.010975204385964565	0.018738870200625313	0.000092519431352912644
-0.013587755738108276	0.014982198281539464	-0.00018495293315187017
-0.015435600216143475	0.010420578275316256	0.000097931011619376679
-0.016329268067457929	0.0053998161373836669	-0.00032614311638072611
-0.016138999035742615	0.00028164113707657363	0.00015551639822265951
-0.014804899842927034	-0.0045833416195247894	-0.00048552399712837289
-0.012350028710068326	-0.008880435133578099	-0.00080811033075463973

## IV. Annex IV

### 1. Tasques a realitzar

- a) Descripció del fonament teòric
- b) Adquisició dels materials necessaris
- c) Muntatge i connexions dels sensors, el microcontrolador i l'ordinador.
- d) Elaboració del software del microcontrolador i de Matlab.
- e) Captura i emmagatzematge de les dades
- f) Anàlisi previ per al tractament de la senyal
- g) Elecció de la freqüència de mostreig. Anàlisi d'espectre:
- h) Disseny de filtres FIR (Resposta Impulsional Finita)
- i) Monitorització de les dades de vibracions
- j) Anàlisi de la senyal (elecció de valors òptims per al càlcul)
- k) Interpretació de dades (Càlcul del període, l'amplitud i la fase de l'acceleració)
- l) Càlcul dels contrapesos per a l'equilibrat (massa i posició)
- m) Interfase de presentació de resultats a l'ordinador.
- n) Determinar la precisió assolible, les vibracions residuals i les seves causes
- o) Recomanacions per a un manual d'operari

## 2. Pressupost

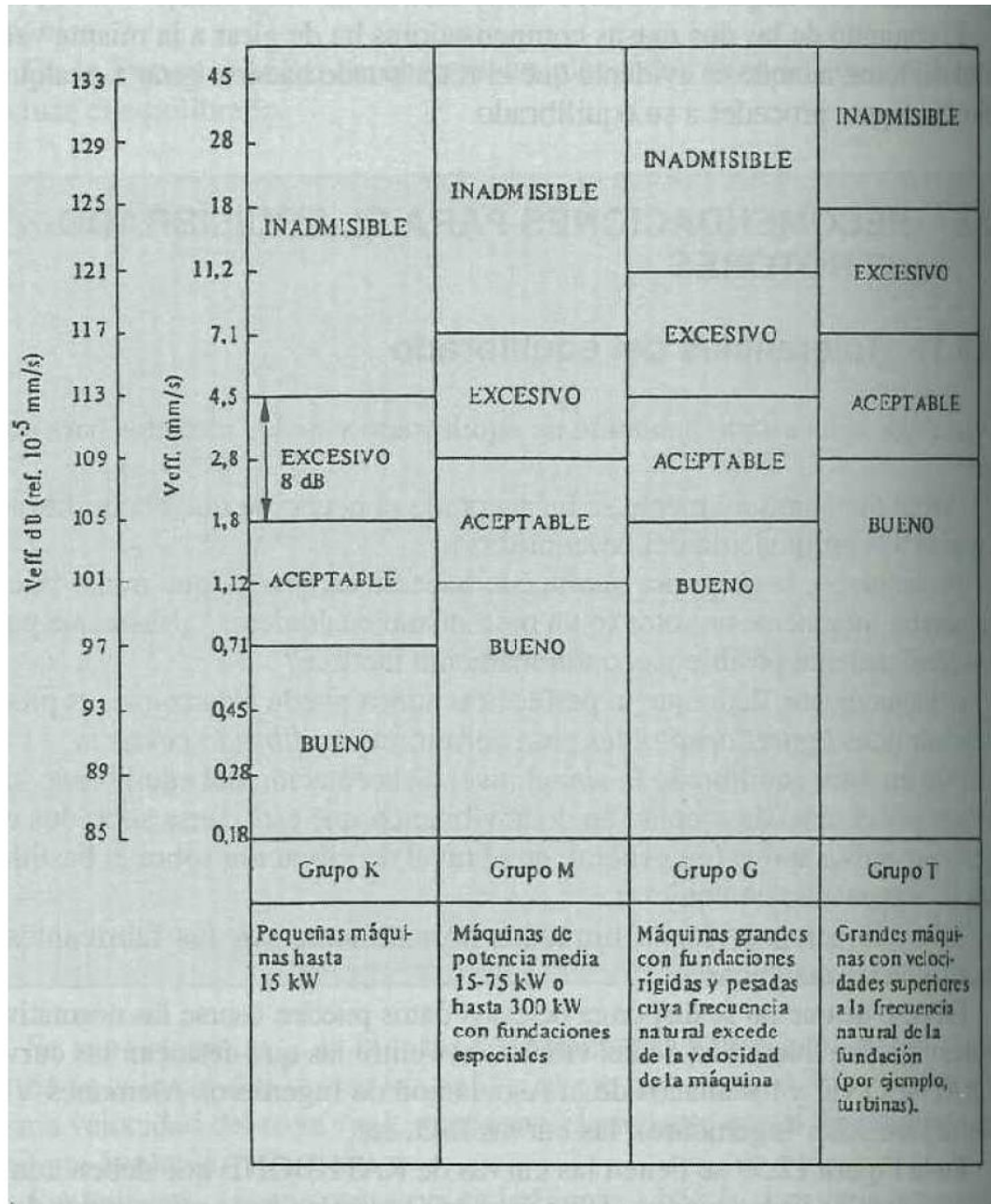
<b>MATERIAL:</b>	<b>Descripció</b>	<b>Preu unitari</b>	<b>Quantitat</b>	<b>Import</b>
	Microcontrolador STM32F4 Discovery	18,15	1	18,15
	Acceleròmetre MPU6050	5,45	1	5,45
	Transductor òptic CNY7	3,20	1	3,20
	Connexió USB - RS232	6,20	1	6,20
	Connexió USB - $\mu$ C	3,70	1	3,70
	Cables connexions	17,50	(diversos tipus)	17,50
	Petita electrònica (R, C, BC547, NE555, Led, placa)	2,50	(material divers)	2,50
<b>TOTAL</b>				<b>56,70 €</b>

<b>MÀ D'OBRA: Tasca</b>	<b>Duració prevista (setmanes)</b>	<b>Duració real (setmanes)</b>	<b>Hores</b>
I. Cerca i estudi de bibliografia.	4	4	80
II. Treball previ ( $\mu$ C, sensors, ventiladors)	6	6	120
III. Adquisició i muntatge dels materials	6	8	30
IV. Software del $\mu$ C i Matlab per comunicació	2	2	40
V. Definir el procediment d'equilibrat	1	4	80
VI. Redacció del projecte	2	2	60
VII. Elaboració de la presentació	1	1	20
<b>TOTAL</b>	<b>22</b>	<b>27</b>	<b>430</b>

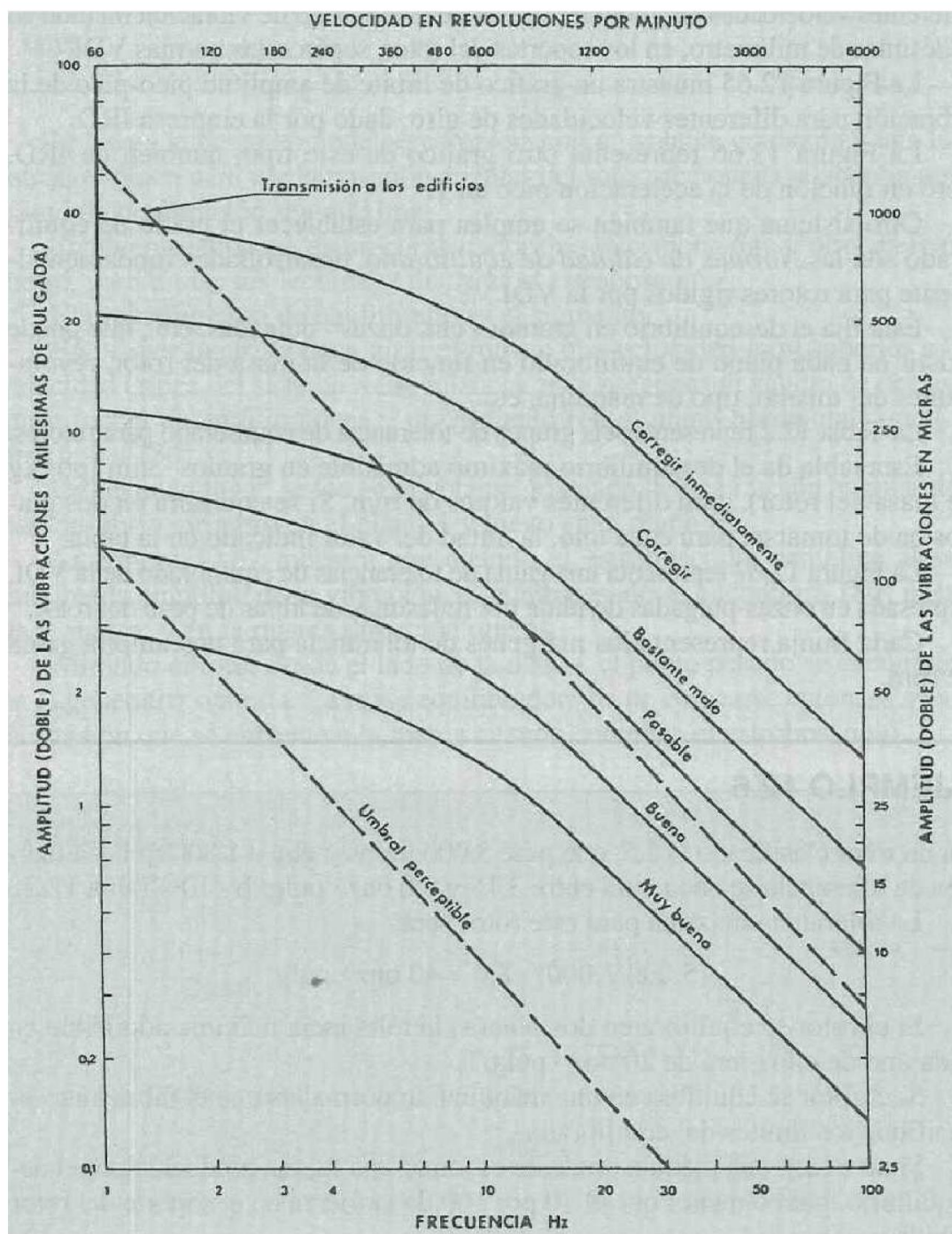


### 3. Taules de toleràncies a vibracions residuals

Taula 4: Toleràncies vibracions en funció del tipus de màquina



Taula 5: Toleràncies vibracions en funció de la freqüència de rotació



Taula 6: Límits de vibració segons la norma ISO 10816-1

A – máquina nueva, B – normal, C – no utilizable, D – inadmisible

Effektivwert der Schwinggeschwindigkeit mm/s	Klasse I	Klasse II	Klasse III	Klasse IV
0,28	A	A	A	A
0,45				
0,71				
1,12	B	B	B	B
1,8				
2,8	C	C	C	C
4,5				
7,1	D	D	D	D
11,2				
18				
28				
45				

**Clase I:**Máquinas pequeñas hasta 15 kW,  
Especialmente Electromotores**Clase II:**Máquinas medianas, 15 kW a 75 kW,  
Sin cimentación especial**Clase III:**Grandes generadores y máquinas sobre  
Cimentaciones pesadas.**Clase IV:**Grandes generadores y máquinas sobre  
cimentaciones aisladas.  
(Turbomáquinas)

## V. Indicacions per a l'operari

### 1. Consideracions i comprovacions prèvies a l'equilibrat

- a) Verificar que la màquina està correctament instal·lada, sense peces soltes al rotor ni cargols trencats o fluixos.
- b) Verificar que l'ancoratge sigui el correcte, sense fissures a la cimentació.
- c) Tots els cicles de mesura es faran amb el mateix règim i sentit de gir. Petits canvis en el règim poden causar grans variacions de magnitud i fase en les vibracions. Tampoc es modificarà la posició de cap sensor en els diferents cicles de mesura.
- d) Evitar velocitats de rotació properes a la freqüència natural de la màquina per evitar ressonàncies.
- e) Conèixer la velocitat crítica de la màquina o estar segurs que no hi arribarem.
- f) Caldrà tenir accés al rotor per aplicar les masses d'assaig i compensació.
- g) El pes de prova haurà de generar una variació de l'acceleració en angle i fase entre el 30 i el 100% de la vibració original. Pesos petits causen errors i pesos grans poden provocar vibracions perilloses o fer reaccionar la màquina amb rigideses diferents, perdent linealitat respecte a la reacció original.
- h) L'acceleròmetre haurà de quedar correctament fixat (generalment amb un imant). Haurà de d'estar el màxim d'horitzontal possible per captar tot el moviment en la direcció que volem mesurar. Es situarà el més prop possible del pla de rotació.
- i) Col·locar el sensor òptic de manera que coincideixi amb sentit de l'eix X positiu de l'acceleròmetre.
- j) Col·locar un reflectant en una de les pales de manera que sigui visible al sensor. Girar manualment el ventilador per comprovar que el sensor detecta el reflectant (LED vermell del sensor encès). Comprovar que el sensor no detecta les aspes sense reflectant. Si cal, ajustar la distància del sensor a l'aspa
- k) Si equilibrem en un pla, el disc no hauria de tenir enguerximent seccional (també conegut per alabeig). En el cas d'un ventilador comprovar que els aspes estan totes a la mateixa distància del sostre. Rectificar la posició de les aspes que no ho estiguin.

### 2. Procediment d'equilibrat

- 1) Mesurar la vibració inicial causada pel desequilibri.
- 2) Aplicar un pes de prova i tornar a mesurar la vibració.
- 3) Comprovar que ha variat la vibració entre un 30 i un 100%. En cas negatiu repetir el pas 2 amb un pes de prova més idoni.
- 4) Retirar el pes de prova i aplicar els contrapesos de correcció.
- 5) Mesura de la vibració residual. Si és excessiva mirar de reajustar els contrapesos de correcció.

### 3. Erros freqüents

- a) No retirar el pes de prova.
- b) Pes de prova insuficient (magnifica els errors de mesura) o excessiu (causa no linealitat).
- c) Aplicació de la correcció en posició angular incorrecta (per error en el maneig del equip de mesura)
- d) Mesurament de velocitat angular inestable, degut a la manca de contrast de la marca al disc que ha de llegir el sensor o a un ajust incorrecte.